



**Высшая
проба**

ВСЕРОССИЙСКАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ

РЕКОМЕНДАЦИИ ДЛЯ УЧАСТНИКОВ

Всероссийской олимпиады школьников «Высшая проба»
по профилю «Промышленное программирование» для 9, 10, 11 класса

2024/2025 уч. г.



ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Об олимпиаде.....	3
О подготовке к олимпиаде	4
Структура олимпиадного задания	5
Теория.....	5
Алгоритмы.....	5
Данные.....	5
Бэкенд	5
Фронтенд	6
Темы для подготовки.....	7
Теория.....	7
Алгоритмы.....	7
Данные.....	8
Frontend.....	9
Backend	9
Литература и источники для подготовки	10
Теория.....	10
Алгоритмы.....	10
Данные.....	11
Frontend.....	11
Backend	11
Языки программирования, используемые в тестирующей системе	13
Дополнительные фреймворки, установленные в тестирующей системе ...	13

Об олимпиаде

Олимпиада «Высшая проба» по промышленному программированию представляет собой уникальную возможность для школьников, которые стремятся поступить в топовые вузы на лучших условиях и построить успешную карьеру в IT-индустрии. Олимпиада проводится в два этапа и охватывает задачи как на теоретические знания, так и на практические навыки.

Первый этап проходит в режиме онлайн и включает в себя задачи на знания теории, алгоритмы, работу с данными, а также задания по фронтенд- и бэкенд-разработке. Участники, которые успешно пройдут первый этап, будут приглашены на второй этап, который состоится очно на площадках по всей России. На этом этапе вас ждут более сложные задачи, приближенные к реальным кейсам из индустрии.

О подготовке к олимпиаде

Подготовка к олимпиаде по промышленному программированию требует комплексного подхода. Прежде всего, это знание теории, алгоритмов и структур данных, которое будет проверяться на олимпиаде. Для успешного участия также необходимо владение одним из языков программирования (Python, Go, C++ или Java), навыками работы с данными (JSON, XML, CSV), базами данных и SQL, а также знание основ фронтенд-разработки (HTML, CSS, JavaScript) и протоколов интернет-связи (HTTP, TCP, UDP).

Особое внимание стоит уделить практике: регулярное решение прикладных задач и прохождение профильных курсов или участие в летних школах помогут развить навыки, необходимые для успешного участия в олимпиаде. Для подготовки можно познакомиться с демонстрационным вариантом и образовательными ресурсами, которые помогут вам освоить необходимые технологии и подходы.

Не забывайте, что каждая задача — это шаг к достижению вашей мечты. Успешное участие в олимпиаде откроет вам двери в ведущие университеты страны и станет важным шагом на пути к карьерным достижениям в IT.

Структура олимпиадного задания

Теория

2 задания, максимальная оценка за каждое задание — 5 баллов

- Ответы вводятся текстом
- Не требуется писать код
- Знание теории информации, кодирования, представления чисел в различных системах счисления, измерения информации, файловой системы компьютера, IP-адресации

Алгоритмы

3 задания, максимальная оценка за каждое задание — 10 баллов

- Решение в виде кода на разрешенном языке программирования
- Цель: написать оптимальный код
- Неоптимальные решения получают частичный балл
- Похоже на задачи классических олимпиад по информатике

Данные

3 задания, максимальная оценка за каждое задание — 10 баллов

- Работа с данными, представленными в различных форматах (JSON, CSV, XML), а также с реляционными БД
- Решения могут быть представлены SQL-запросами или однозначными ответами, полученными из представленных данных
- Решение одной из задач может быть получено любым способом: через электронные таблицы, код, анализ баз данных и текстовых файлов
- Необходимо понимание организации файловой системы и её обход вручную или программно

Бэкенд

2 задания, максимальная оценка за каждое задание — 10 баллов

- Создание веб-сервисов
- Технологии практически не ограничены (указаны в условиях задачи)

- Автоматическое тестирование решений

Фронтенд

1 задание, максимальная оценка за задание — 10 баллов

- Навыки в вёрстке и JavaScript
- Использование HTML/CSS для решения задач
- Частичное соответствие результата также оценивается

Темы для подготовки

Теория

- Измерение информации
- Системы счисления
- Кодирование информации
- Алгоритмы для исполнителей
- Логические основы компьютерной техники
- Математические модели
- Файловая система
- Компьютерные сети

Алгоритмы

- Базовый синтаксис высокоуровневого языка программирования
- Переменные, типы, выражения
- Базовый ввод-вывод
- Условные конструкции, логические выражения
- Циклы
- Функции, передача параметров, рекурсия

- Базовые типы данных (целый, вещественный, логический, символьный)
- Массивы (в т. ч. многомерные) и их обработка
- Строки и их обработка
- Графы и деревья

- Представление об асимптотической сложности алгоритмов
- Навыки оценки применимости алгоритма для заданных ограничений размера входных данных, времени и памяти

- Решения на основе полного перебора (итеративного или рекурсивного)
- Жадные алгоритмы
- Разделяй-и-властвуй
- Динамическое программирование

- Свойства целых чисел (четность, делимость и др.)

- Дроби, проценты
 - Системы счисления
 - НОД, НОК
 - Простые числа, решето Эратосфена
 - Быстрое возведение в степень
 - Базовая геометрия на плоскости
-
- Линейный поиск в массиве
 - Двоичный поиск
 - Алгоритмы сортировки со сложностью $O(N \log N)$
-
- Стек и очередь
 - Двоичная куча
 - Двоичные деревья поиска
 - Хеш-таблицы
-
- Основы теории графов (способы задания графа, основные свойства графов)
 - Обходы деревьев
 - Обход графа в глубину и его применения
 - Поиск кратчайших путей в графах

Данные

- Файловый ввод-вывод
- Сортировка и фильтрация данных во внутренней памяти по произвольным критериям
- Чтение, запись и обработка текстовых форматов хранения и обмена данными (CSV, JSON)
- Обработка данных в электронных таблицах (Excel, OpenOffice Calc)
- Обработка данных и анализ данных с помощью различных библиотек
- Файловая система компьютера, библиотеки для работы с файлами и папками
- Выполнение запросов на языке SQL к многотабличной базе данных, представленной в формате файловой СУБД SQLite

- Проектирование реляционной модели данных на диалекте СУБД SQLite в следующем объеме: таблицы, первичные и внешние ключи, индексы, представления данных

Frontend

- Язык HTML и каскадные таблицы стилей CSS
- Язык программирования JavaScript
- Основы блочной верстки

Backend

- Клиент-серверная архитектура
- Обработка HTTP-запросов
- Сетевое взаимодействие и потенциальные проблемы
- Обработка ошибок
- Применение простых алгоритмов на практике

Литература и источники для подготовки

Теория

- УМК «Информатика» К.Ю. Полякова, Е.А. Еремина, 10-11 классы. Базовый и углубленный уровни. <https://lbz.ru/books/742/>
- Угринович, Н. Информатика и информационные технологии / Н. Угринович. — М.: Бином. Лаборатория знаний, 2017. — 512 с.
- <https://kpolyakov.spb.ru/>
- <https://inf-ege.sdamgia.ru/>

Алгоритмы

- Алгоритмы: построение и анализ / Томас Х. Кормен, Чарльз И. Лейзерсон, Рональд Л. Ривест, Клиффорд Штайн. — 3-е изд. — М.: Вильямс, 2013. — 1328 с.
- Дасгупта С., Пападимитриу Х., Вазирани У. Алгоритмы / С. Дасгупта, Х. Пападимитриу, У. Вазирани. — М.: МЦНМО, 2014. — 320 с.
- Лааксонен А. Олимпиадное программирование. / Антти Лааксонен. — 2-е изд. — М.: ДМК Пресс, 2020. — 328 с.
- Московские олимпиады по информатике / Под ред. Е. В. Андреевой, В. М. Гуровица и В. А. Матюхина. — М.: МЦНМО, 2006. — 256 с.
- Халим С., Халим Ф. Спортивное программирование / пер. с англ. Н. Б. Желновой, А. В. Снастина. — М.: ДМК Пресс, 2020. — 604 с.
- <https://e-maxx.ru>
- <https://neerc.ifmo.ru/wiki/index.php>
- <https://brestprog.by/topics>
- <https://ru.algorithmica.org>
- <https://acm.khpnets.info>
- <https://informatics.msk.ru>
- <https://codeforces.com>
- <https://acmp.ru>

Данные

- Маккинни У. Python и анализ данных. — 3-е изд. — М.: ДМК Пресс, 2023. — 537 с
- SQL учебник для чайников: <https://html5css.ru/sql/default.php>
- Самоучитель SQL: <https://practicum.yandex.ru/blog/samouchitel-sql/>

Frontend

- Резиг, Бибо, Марас: Секреты JavaScript ниндзя
- Дэвид Фленаган: JavaScript. Полное руководство. 7-е издание.
- Современный учебник JavaScript <https://learn.javascript.ru/>

Backend

- Clean Architecture by Rober Martin
- Clean Code by Rober Martin

- Core Python Programming by Wesley Chun
- Effective Python by Brett Slatkin
- Learning Python by Mark Lutz

- The Go Programming Language by Alan A. A. Donovan and Brian W. Kernighan
- Effective Go by the Go Team

- Interconnections by Radia Perlman

- Site Reliability Engineering: How Google Runs Production Systems, edited by Betsy Beyer, Chris Jones, Jennifer Petoff, and Niall Murphy
- The Site Reliability Workbook, edited by Betsy Beyer, Niall Murphy, David Rensin, Kent Kawahara, and Stephen Thorne
- Google Hacks, Tara Calishain and Rael Dornfest
- Real World SRE, Nat Welch
- Implementing Service Level Objectives, Alex Hidalgo

- The C++ Programming Language (4th Edition), Bjarne Stroustrup
- The C++ Standard Library, Nicolai M. Josuttis
- Python Web Development with Django by Forcier, Bissex, and Chun

- Design Patterns, Erich Gamma, et al.
- Refactoring to Patterns (Addison-Wesley Signature Series), Joshua Kerievsky

- Security Engineering, Ross Anderson
- Building Secure & Reliable Systems, Heather Adkins, Betsy Beyer, Paul Blankinship, Ana Oprea, Piotr Lewandowski, Adam Stubblefield

- How Google Tests Software, Whittaker

- Programming with POSIX(R) Threads, David R. Butenhof

Языки программирования, используемые в тестирующей системе

1. Python 3.12
2. Go 1.23
3. C++ 20
4. Java 17
5. Node.js 20.14
6. Браузер Chrome 125 (для проверки задач на HTML/CSS/JS)

Дополнительные фреймворки, установленные в тестирующей системе

1. Для языка **Python** доступны библиотеки:
 - a. Flask
 - b. aiohttp
 - c. FastAPI
2. Для языка **Java** доступны библиотеки:
 - a. json-simple
 - b. jackson
 - c. async-http-client
3. Для языка **C++** доступна библиотека **httplib.h**