

Утверждено
Распоряжением министерства
образования Иркутской области
от «14» *февраля* 2025 года № *55-168-149*

ПОЛОЖЕНИЕ
об областной олимпиаде по программированию
среди студентов, обучающихся по специальностям
и профессиям среднего профессионального образования укрупненной
группы 09.00.00 «Информатика и вычислительная техника»

I. ОБЩИЕ ПОЛОЖЕНИЕ

1.1. Настоящее Положение определяет порядок проведения областной олимпиады по программированию среди студентов, обучающихся по специальностям и профессиям среднего профессионального образования укрупненной группы профессий и специальностей 09.00.00 Информатика и вычислительная техника (далее – Олимпиада), ее организационное и методическое обеспечение, порядок участия в Олимпиаде и определения победителя и призеров.

1.2. Оператором по подготовке и проведению Олимпиады является Государственное бюджетное профессиональное образовательное учреждение Иркутской области «Иркутский авиационный техникум» (далее – ИАТ).

II. ЦЕЛИ И ЗАДАЧИ ОЛИМПИАДЫ

2.1. Олимпиада проводится в целях выявления качества подготовки выпускаемых специалистов, дальнейшего совершенствования их мастерства, закрепления и углубления знаний и умений, полученных в процессе теоретического и практического обучения, стимулирования творческого роста, повышения престижа среднего профессионального образования, выявления наиболее одаренных и талантливых обучающихся в области информационных технологий и посвящена 90-летию Государственного бюджетного профессионального образовательного учреждения Иркутской области «Иркутский авиационный техникум».

2.2. Основными задачами Олимпиады являются:

- повышение интереса студентов к своей будущей профессии, осознание её социальной значимости;
- стимулирование творческого роста студентов, выявление одаренной молодежи для формирования кадрового потенциала для исследовательской и производственной деятельности России в сфере высоких технологий;
- выявление уровня подготовки студентов, закрепление и углубления знаний и умений студентов, полученных в процессе теоретического, практического и профессионального обучения;
- проверка способностей студентов к системному действию в профессиональной ситуации;

- проверка готовности будущего специалиста к самостоятельной трудовой деятельности;
- совершенствование навыков самостоятельной работы и развития профессионального мышления;
- обмен опытом в области профессионального обучения по новым технологиям.

III. ОРГАНИЗАЦИЯ И ПРОВЕДЕНИЕ ОЛИМПИАДЫ

3.1. Олимпиада проводится 28 марта 2025 года на базе ИАТ по адресу: г. Иркутск, ул. Ленина, 5 «а».

3.2. К участию в Олимпиаде допускаются студенты, обучающиеся по специальностям и профессиям среднего профессионального образования укрупненной группы профессий и специальностей 09.00.00 «Информатика и вычислительная техника», прошедшие регистрацию на сайте or.irkat.ru в срок до 21 марта 2025 года.

3.3. Процедуру регистрации участников Олимпиады от образовательного учреждения осуществляет ответственный сотрудник (руководитель команды), назначенный приказом руководителя.

3.4. Полный список участников Олимпиады публикуется на сайте олимпиады (or.irkat.ru) 21 марта 2025 года.

3.5. От одной образовательной организации в Олимпиаде может принимать участие не более 3 человек.

3.6. Руководитель команды от образовательной организации входит в состав жюри.

3.7. Организацию работы по подготовке и проведению Олимпиады осуществляет оргкомитет Олимпиады, состав которого утверждается приказом руководителя ИАТ.

3.8. В функции оргкомитета входят: разработка плана подготовки и проведения Олимпиады, разработка конкурсных заданий и критериев их оценки.

3.9. План проведения Олимпиады:

Время	Место	Мероприятие
09:00 – 09:30	Холл актового зала ИАТ (3 этаж)	Регистрация участников Олимпиады и их руководителей
09:30 – 09:50	Актовый зал ИАТ	Открытие олимпиады, инструктаж
09:50 – 14:00	Аудитории ИАТ: 206, 208, 209, 211	Проведение Олимпиады
10:00 – 11:00	Аудитория ИАТ 315	Согласование критериев оценки
11:00 – 13:00	Аудитория ИАТ 315	Круглый стол (315 ауд.) 1. «Демонстрационный экзамен: из опыта»

		2. «Дидактическое содержание профессиональных дисциплин и МДК»
13:00 – 13:30	Буфет	Обед
13:10 – 14:00	Аудитория ИАТ 315	Проверка заданий
14:00 – 15:00	Актовый зал ИАТ	Подведение итогов, награждение

3.10. Программное обеспечение: для выполнения заданий Олимпиады всем участникам предлагаются (на выбор) следующие программные продукты:

- Pascal ABC.NET;
- Visual Studio 2022 (C#, VB.NET, C++);
- Java 21;
- Python 3;
- Go 1.21;
- Node.js 20.

IV. ПРАВИЛА УЧЕТА РЕШЕНИЙ

4.1. Решения участников Олимпиады будут оцениваться строго по формальному принципу с помощью тестирования на некотором множестве тестов. Поэтому к решениям участников и их оформлению предъявляются жесткие формальные требования, за несоответствие которым жюри оставляет за собой право оценивать решения в 0 (ноль) баллов.

4.2. Задания по программированию предполагают решение в виде написания программы на выбранном языке программирования. Разработанные программы должны выполнять обработку каких-либо входных данных согласно условию задачи.

4.3. Тесты для задачи представляют собой наборы входных данных согласно условию задачи, а также тестирующий модуль, который для конкретного набора входных данных (теста) может проверить правильность набора выходных данных, полученных с помощью программы-решения участника Олимпиады. Для каждого олимпиадного задания разрабатывается свой набор тестов.

4.4. Оценка решения участника производится в зависимости от количества пройденных решением тестов. Тест считается пройденным, если для данного теста с помощью решения участника получены выходные данные, правильные согласно условию задачи. Каждому тесту сопоставляется какое-то количество баллов за прохождение теста в зависимости от сложности данного теста. Таким образом, каждое решение оценивается в сумму баллов за пройденные данным решением тесты.

4.5. Форматы входных файлов строго соответствуют заявленным в условиях задач. Форматы выходных файлов решения должны строго соответствовать требуемым форматам в условиях задач. Любое несоответствие

может быть рассмотрено жюри в качестве неверного решения с оценкой в 0 (ноль) баллов.

4.6. Решения, которые являются «подгонкой» под правильный ответ с целью пройти ряд тестов, оцениваются жюри в 0 баллов независимо от того, сколько такое решение набрало баллов в результате тестирования. Такие решения выявляются жюри при изучении исходного кода программ. Если к решению участника исходных текстов не прилагается, то такое решение оценивается в 0 (ноль) баллов.

V. ПРАВИЛА ПОВЕДЕНИЯ УЧАСТНИКОВ ОЛИМПИАДЫ И РУКОВОДИТЕЛЕЙ КОМАНД ПРИ ВЫПОЛНЕНИИ ЗАДАНИЙ

5.1. Запрещаются переговоры между участниками Олимпиады (далее – участник), между участниками и руководителями команд.

5.2. Строго запрещается пользоваться сотовыми телефонами, справочниками и другими источниками информации (за исключением источников, инсталлированных на ПК).

5.3. Участник может покинуть аудиторию только в сопровождении волонтера.

5.4. Участник обязательно должен согласовать свой выход за пределы техникума со своим руководителем.

5.5. Руководитель команды имеет право заходить в аудитории, где участники выполняют задания, но при этом контакт с участниками в той или иной форме запрещён.

VI. ПРАВИЛА ОФОРМЛЕНИЯ РЕШЕНИЙ

6.1. Учитывая, что решения участников Олимпиады оцениваются с помощью тестов, при разработке решений следует строго соблюдать форматы входных и выходных данных, указанных в условии задачи

6.2. В общем случае программа-решение состоит из трех частей:

- чтение входных данных из входного файла;
- обработка данных для поиска правильного ответа;
- запись полученного ответа в выходной файл.

6.3. Примеры оформления решений представлены в Приложении к данному положению.

VII. КРИТЕРИИ ОЦЕНИВАНИЯ

Критерии оценивания согласовываются на заседании оргкомитета.

VIII. ПОДВЕДЕНИЕ ИТОГОВ ОЛИМПИАДЫ

8.1. Для подведения итогов Олимпиады и определения победителей и призеров (второе и третье место) оргкомитетом Олимпиады создается жюри.

8.2. В состав жюри входят представители от образовательных организаций, направивших студентов для участия в Олимпиаде, и ИАТ.

8.3. По итогам Олимпиады определяются победитель и призеры среди участников, а также победитель и призеры среди команд. Победитель и призеры среди команд определяются в случае, если численность команды составляет три человека.

8.4. Победителем Олимпиады считается участник (команда участников), набравший наибольшее количество баллов, призерами – участники (команды участников), набравшие по количеству баллов второй и третий результаты.

8.5. По результатам работы жюри составляет протокол, в котором указываются победители и призеры Олимпиады.

8.6. Победители и призеры Олимпиады награждаются соответствующими дипломами.

8.7. Участникам Олимпиады вручаются сертификаты участников.

8.8. Образовательным организациям, подготовившим победителей и призеров, вручаются благодарственные письма.

Приложение
к Положению об областной
олимпиаде по программированию
среди студентов, обучающихся по
специальностям и профессиям
среднего профессионального
образования укрупненной группы
09.00.00 «Информатика и
вычислительная техника»

Примеры оформления решений задачи «Сумма чисел» на различных языках программирования

Пример задания

Имя входного файла:	Input.txt
Имя выходного файла:	Output.txt
Максимальное время работы на одном тесте:	3 секунды
Максимальный объем используемой памяти:	50 мегабайт
Максимальная оценка	100 баллов

Задача (сумма чисел):

Требуется написать программу, которая складывает два целых числа.

Формат входных данных

Входной файл содержит одну строку, в которой через пробел представлены два целых числа.

Формат выходных данных

Выходной файл должен содержать целое число. Гарантируется, что решение существует.

Примеры входных и выходных данных

Input.txt	Output.txt
12 3	15
3 -5	-2
0 9	9
-2 -3	-5

Примеры решения задачи

Pascal ABC.NET	Visual C#
<pre> const InputFileName = 'input.txt'; OutputFileName = 'output.txt'; var A, B, C: Integer; Input, Output: Text; begin Assign(Input, InputFileName); </pre>	<pre> using System ; using System.IO ; using System.Text ; namespace ConsoleApplication2 { class Program { static void Main (string[] args) { </pre>

<pre> Reset(Input); Assign(Output, OutputFileName); Rewrite(Output); ReadLn(Input, A, B); C := A + B; WriteLn(Output, C); Close(Input); Close(Output); end. </pre>	<pre> StreamReader input = new StreamReader("input.txt"); String[] line = input.ReadLine().Split(' '); int a = Convert.ToInt32(line[0]); int b = Convert.ToInt32(line[1]); int c = a + b; StreamWriter output = new StreamWriter("output.txt"); output.Write(c); output.Close(); } } } </pre>
Visual C++	Visual Basic .NET
<pre> #include "stdafx.h" #include <iostream> #include <fstream> using namespace std ; int main () { ifstream input ("input.txt"); int a, b, c; input >> a >> b; c = a + b; std::ofstream out("output.txt"); out << c; out.close(); return 0; } </pre>	<pre> Imports System.IO Module Module1 Sub Main () Dim line As String , parts As Array, a As Integer , b As Integer , c As Integer Using input As New StreamReader("input.txt") line = input.ReadLine() End Using parts = line.Split(" ") a = parts(0) b = parts(1) c = a + b Using output As New StreamWriter("output.txt") output.Write(c) End Using End Sub End Module </pre>
Java	Python
<pre> import java.util.*; import java.io.*; public class Main{ public static void main(String[] argv) throws IOException{ new Main().run(); } PrintWriter pw; Scanner sc; public void run() throws IOException{ sc = new Scanner(new File("input.txt")); int a=sc.nextInt(), b=sc.nextInt(); pw = new PrintWriter(new File("output.txt")); pw.print(a+b); pw.close(); } } </pre>	<pre> fin = open("input.txt") fout = open("output.txt", "w") a, b = map(int, fin.readline().split()) fout.write(str(a+b)) fin.close() fout.close() </pre>
Go	Node.js
<pre> package main import («fmt»; «os») func main(){ in, _ := os.Open(«input.txt») out, _ := os.Create(«output.txt») var a, b int fmt.Fscanf(in, «%d %d», &a, &b) fmt.Fprintf(out, «%d», a+b) } — js import («fmt»; «os») func main(){ in, _ := os.Open(«input.txt») out, _ := os.Create(«output.txt») var a, b int fmt.Fscanf(in, «%d %d», &a, &b) fmt.Fprintf(out, «%d», a+b) } — js </pre>	<pre> const fs = require('fs'); const readableStream = fs.createReadStream('input.txt', 'utf8') readableStream.on('data', (data) => { let input = data.split(' '); throw data; let a = +input[0]; let b = +input[1]; let c = a+b; fs.writeFileSync(«output.txt», c.toString()); }) </pre>