

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>