

Требования
к порядку выполнения заданий школьного этапа
Республиканской предметной олимпиады обучающихся общеобразовательных
организаций Донецкой Народной Республики
в 2022-2023 учебном году
по информатике

1. Откройте папку соответствующего класса.
2. Откройте файл с заданиями, скачайте и просмотрите их.
3. Время выполнения заданий – 3 астрономических часа.
4. Задания для 5-6 классов выполняются в виде записи решений или пояснений и ответов.
5. Задания для 7-11 классов выполняются практически на языке программирования. Допускается выполнение заданий в виде алгоритма или модели решения задачи в любой форме.
6. После выполнения работы файлы работы отправьте по ссылке [«Отправить»](#) на сайте школы. Если задание выполнялось на языке программирования, высылаются файлы с исходным кодом программ (.pas, .cpp или .py)
7. В случае непредвиденных ситуаций обратитесь к учителю информатики и ИКТ Лысенко Сергею Борисовичу по телефону 071-305-30-85.

Памятка для участников,
выполняющих задание на языке программирования

1. Внимательно **прочитайте условия всех задач.**
 2. **Решайте** первой ту задачу, которая вам кажется **самой простой**. Как правило, задачи даются в порядке увеличения сложности, но это лишь субъективная оценка членов жюри.
 3. Не забывайте **регулярно сохранять** промежуточные версии программ. Время, затраченное на перезагрузку компьютера и на повторное написание несохраненной версии программы, не компенсируется.
 4. **Комментарии** в тексте программы **необязательны**. Оценивание осуществляется по результатам запуска на тестовых наборах данных. Оценивается **правильность и эффективность решения**. **Текст программы-решения не оценивается.**
 5. Старайтесь составить наиболее **оптимальный алгоритм решения задачи**, поскольку время, которое будет даваться вашей программе на прохождение каждого теста, ограничено.
 6. Если программа **неверно работает на примере из условия**, то скорее всего и при проверке в системе получит 0 баллов. Если **программа не компилируется** в проверяющей системе, **посмотрите сообщения об ошибках в протоколе и исправьте их.**
 7. **Протестируйте** вашу программу самостоятельно на подготовленных вами контрольных примерах.
- Проверяйте правильность** работы вашей программы **не только на примере из условия**. Особое внимание уделите так называемым «крайним случаям» (например, когда какая-либо величина из условия равна своему наименьшему или наибольшему допустимому значению).
8. В случае, если вы **не можете написать полное решение задачи**, постарайтесь написать **частичное решение**, то есть решение, которое будет давать верный результат в каком-то частном случае или при небольших исходных данных. Это позволит набрать вашей программе хотя бы какие-то баллы. **В задачах с частичным решением:** если вы

рассчитываете получить только часть баллов, внимательно **прочитайте, за что именно начисляется часть баллов** и что при этом нужно выводить.

14. **Решение задачи** – это **один** файл, содержащий **текст программы** на одном из языков программирования **Pascal, C++** или **Python** (другие языки недопустимы). Только этот файл (.pas, .cpp или .py) сдается в жюри олимпиады.

15. Программа должна компилироваться.

16. Программа должна считать данные со **стандартного устройства ввода** (клавиатура), обработать его, вывести на **стандартное устройство вывода** (экран) полученные результаты и завершить **корректно** свою работу.

17. Программа не должна самостоятельно использовать **никаких файлов**.

18. **Проверять корректность данных** не требуется. Гарантируется, что они всегда удовлетворяют указанным в задаче ограничениям и записаны в том формате, который указан в условии (см. раздел «Входные данные»). Помните, что каждая строка *заканчивается* переводом строки, а элементы в строке *отделяются* друг от друга ровно одним пробелом (если в условии не указано обратное).

19. **Результатом работы программы** считается все то, что программа выводит на стандартное устройство вывода (с помощью write(-ln), printf, cout, print), поэтому **ни в коем случае не выводите подсказок** («Введите N», «Сумма элементов будет равна ...» и т.п.).

20. **Порядок и формат вывода результатов** описан в условии задачи (см. раздел «Выходные данные»). Строго придерживайтесь его.

21. **Ни в коем случае не используйте прямой консольный ввод/вывод!** Поэтому ни при каких обстоятельствах **не подключайте** модуль *crt* в Pascal и заголовочный файл *conio.h* в C++, которые его используют.

22. **Не используйте библиотеки и модули**, которые не входят в стандартные дистрибутивы соответствующих компиляторов.

Пример задания

Задание. Написать программу, которая вычисляет сумму элементов массива.

Входные данные. В первой строке задается целое число N ($1 \leq N \leq 30000$) – количество элементов в массиве. Во второй строке записано N чисел, определяющих элементы массива. Все числа целые и по модулю не превосходят 100000.

Выходные данные. Необходимо вывести сумму всех элементов массива.

Пример входных и выходных данных

<i>Ввод</i>	<i>Вывод</i>
5 1 2 3 4 5	15

Пример выполнения задания

Решение на Pascal	Решение на C++	Решение на Python
<pre>var i,N:integer; ai:longint; s:int64; BEGIN s:=0; readln(N); for i:=1 to N do begin read(ai); s:=s+ai end; writeln(s) END.</pre>	<pre>#include <iostream> using namespace std; int main() { int i,N; long ai; long long s=0; cin>>N; for(i=1;i<=N;i++) { cin>>ai; s+=ai; } cout<<s<<endl; return 0; }</pre>	<pre>N=int(input()) a=input().split() s=0 for i in range(N): s+=int(a[i]) print(s)</pre>