

Методические указания для  
подготовки к Олимпиаде  
«Спортивное программирование  
на Урале»

Екатеринбург, 2020 г.

## Об олимпиаде

Олимпиада школьников «Спортивное программирование на Урале» проводится в два этапа. Первый (отборочный) этап проходит в онлайн-формате и состоит из не более 10 задач. Лучшие участники онлайн-тура приглашаются к участию во втором (заключительном этапе), который проводится очно на площадках в разных регионах РФ. Очный этап длится 5 часов и состоит из не более 10 задач.

Решением задачи является файл с исходным текстом программы, написанной на одном из разрешённых языков программирования (C, C++, C#, Pascal, Java, Python). Разные задачи можно решать на различных языках программирования. Допустимый размер исходного текста одной программы с решением ограничен.

Решения участников проверяются с помощью автоматизированной тестирующей системы на заранее подготовленном наборе тестов. Решение засчитывается как верное только при прохождении всех тестов.

## Проверка решений

Входные данные подаются программе в стандартном потоке ввода. Программа должна выводить ответ в стандартный поток вывода.

Чтобы автоматическая проверяющая система смогла протестировать ваше решение, оно должно отвечать следующим требованиям:

- Программа должна являться консольным приложением.
- Входные данные подаются программе в стандартном потоке ввода (ввод с клавиатуры). Программа должна выводить ответ в стандартный поток вывода (вывод на экран).
- Программа должна выводить только те данные, которые требует условие задачи. Выводить приглашение для ввода («Введите N:») не нужно. Также не нужно ожидать нажатия клавиши в конце работы программы.

Входные данные в тестах всегда удовлетворяют ограничениям, описанным в условиях задач. Проверять эти ограничения в своих решениях не требуется.

В решениях задач запрещается:

- работа с любыми файлами;
- выполнение внешних программ и создание новых процессов;
- работа с GUI-элементами (окнами, диалогами и т.д.);
- работа с внешними устройствами (принтером, звуковой картой и т.д.);
- использование сетевых средств.

При проверке решений используются те же компиляторы, что имеются у участников. Однако набор функций, доступных программам во время проверки, ограничен. В ходе пробного тура участники должны убедиться в доступности необходимых им функций. До начала основного тура участник может обратиться к жюри с просьбой о расширении набора доступных функций.

Решение проверяется путём последовательного запуска на наборе тестов, который недоступен участникам и является одинаковым для всех. Тестирование производится автоматически, поэтому программы должны в точности соблюдать форматы входных и выходных данных, описанные в условии каждой задачи.

Для каждой задачи определены максимальное время выполнения, объём доступной памяти и максимальный размер выходных данных для одного теста. Программа участника проходит данный тест, если выдаёт для него верный ответ и укладывается в ограничения по времени, по памяти и по размеру выходных данных.

После отправки очередной задачи на проверку участник получает ответ с результатами её тестирования.

Результатом проверки решения может быть один из следующих вердиктов:

- **Accepted.** Решение засчитано.
- **Compilation error.** Компиляция программы завершилась с ошибкой.  
*Возможные причины:* синтаксическая ошибка; неправильно указан язык.
- **Wrong answer.** Ответ программы неверен. В этом случае сообщается только номер теста. Входные данные теста не предоставляются, т.к. причины ошибки должны быть выявлены автором решения самостоятельно. *Возможные причины:* ошибка в программе; неверный алгоритм; программа выводит ответ в файл.
- **Time limit exceeded.** Программа превысила установленное ограничение по времени. Работа решения будет прервана, как только истечет это

время. *Возможные причины:* бесконечный цикл; неэффективное решение.

- **Memory limit exceeded.** Программа превысила установленное ограничение по памяти. Работа решения будет прервана, как только превышено это ограничение. *Возможные причины:* утечка памяти; неэффективное решение.
- **Output limit exceeded.** Программа превысила ограничение на размер выходных данных. Данное ограничение устанавливается с большим запасом и, как правило, не указывается в условиях задач. *Возможные причины:* бесконечный цикл; ошибка в программе.
- **Idleness limit exceeded.** Программа простаивала в течение длительного промежутка времени. *Возможные причины:* ожидание нажатия клавиши в конце программы; чтение входных данных, когда они уже прочитаны до конца.
- **Runtime error.** Программа аварийно завершила работу. *Возможные причины:* ненулевой код возврата (ошибка «non-zero exit code»); деление на ноль (ошибка «division by zero»); бесконечная рекурсия (ошибка «stack overflow»); недостаточный размер массивов или обращение по недоступному адресу в памяти (ошибка «access violation»).
- **Restricted function.** Программа попыталась совершить небезопасную для проверяющей системы операцию, например, прочитать файл или обратиться к сетевому ресурсу.

Пример: если вы получили вердикт Wrong answer на тесте № 3, то это означает, что ваше решение успешно прошло тесты № 1 и № 2, а на тесте № 3 вывело неверный ответ. Если вы исправили ошибку, отправили решение на проверку снова и получили вердикт Time limit exceeded на тесте № 10, то это означает, что исправленная ошибка действительно проявлялась на тесте № 3. Теперь решение успешно проходит тесты с № 1 по № 9, а на тесте № 10 работает дольше установленного ограничения. При этом неизвестно, был бы ответ на этом тесте правильным в случае, если бы решение уложилось в ограничение по времени.

## Перечень рекомендуемой литературы

- Волчёнков С. Г., Корнилов П. А., Белов Ю. А. и др. Ярославские олимпиады по информатике. Сборник задач с решениями. М.: БИНОМ. Лаборатория знаний. 2010.
- Кирюхин В. М., Окулов С. М. Методика решения задач по информатике. Международные олимпиады. М.: БИНОМ. Лаборатория знаний, 2007.
- Меньшиков Ф. В. Олимпиадные задачи по программированию. СПб.: Питер, 2006.
- Окулов С. М. Программирование в алгоритмах. М.: БИНОМ. Лаборатория знаний. 2002.
- Скиена С. С., Ревилла М. А. Олимпиадные задачи по программированию. Руководство по подготовке к соревнованиям. М.: Кудиц-образ, 2005.

## Список веб-ресурсов

- Сайт УрФУ с архивом задач  
<https://acm.timus.ru/>
- Школа программиста  
<https://acmp.ru/>
- Дистанционная подготовка по информатике  
<https://informatics.mccme.ru/>
- Крупнейшая библиотека описаний и реализаций алгоритмов  
<https://e-maxx.ru/>
- Энциклопедия по теории алгоритмов, составленная студентами ИТМО  
<http://neerc.ifmo.ru/wiki/>
- Сайт с регулярными соревнованиями по программированию  
<https://codeforces.com/>
- Курс по программированию с нуля  
<https://pythontutor.ru/>
- Интернет-энциклопедия по информатике от онлайн-школы «Фоксфорд»  
<https://foxford.ru/wiki/informatika>
- Введение в программирование на C++, онлайн-курс  
<https://stepik.org/course/363/promo>
- Программирование на Python  
<https://stepik.org/course/67/promo>
- Быстрый старт в спортивное программирование  
<https://stepik.org/course/64454/promo>
- Сайт Методического центра олимпиадной информатики  
<http://lbz.ru/metodist/lections/6/>