

Контакты преподавателей:

Крымов Алексей, [СИНИЕ \(WEB-11/ML-11\)](#), krymov@corp.mail.ru, TG [@alkrymov](#)
Корепанов Дмитрий, [ЖЕЛТЫЕ \(WEB-12/ML-12\)](#), dmitrykorepanov@gmail.com, TG [@DKorepanov](#)
Глушенков Дмитрий, [ЗЕЛЕННЫЕ \(WEB-13/ML-13\)](#), d.glushenkov@corp.mail.ru, TG [@gloonix](#)

Итоговая оценка

- Отлично – 84-100 баллов
- Хорошо – 67-83 балла
- Удовлетворительно – 50-66 баллов

Для получения зачета нужно набрать ≥ 50 баллов, решив при этом минимальный набор задач.

Минимальный набор задач

Сдача всех задач из списка ниже является необходимым условием для получения зачета по курсу АиСД.

Модуль 1

- Задача 2 (4 балла)
- Задача 4 (4 балла)
- Задача 6 (3 балла)

Модуль 2

- Задача 1 (6 баллов)
- Задача 2 (4 балла)
- Задача 4 (5 баллов)

Модуль 3

- Задача 1 (5 баллов)
- Задача 2 (3 балла)
- Задача 3 (4 балла)

Правила сдачи задач

1. Соответствие минимальным правилам кодирования (см. ниже)
2. Задача должна соответствовать требованиям, указанным в условии.
3. Задача, загруженная в Я.Контест до его закрытия, может досдаваться студентом неограниченное число раз - на усмотрение преподавателя.

4. Содержание тестов задачи из Я.Контеста можно получить, предъявив свой набор тестов для этой задачи.
5. Задача должна пройти защиту у преподавателя.

Удаленная сдача задач

Для сдачи задач нужно послать своему преподавателю письмо на e-mail, в теме которого указать метку [TPSpring2025], свой поток (WEB/ML), свое имя и фамилию.

В теле письма нужно указать:

- оперативный контакт
- ссылки на посылки в конкурсе, которые вы хотите сдавать.

На одну сдачу - один тред в почте.

Пример письма:

Тема: [TPSpring2025] WEB Иванов Иван

Тело письма:

Телега: @adfafadsf

Прошу принять задачи

Задача 1 вариант 1 <https://contest.yandex.ru/contest/17460/run-report/31591176/>

Задача 2 вариант 4 <https://contest.yandex.ru/contest/17460/run-report/31591176/>

Ссылка на посылку находится здесь:

Отправить		i осталось 99 попыток								
Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
17 мрт 2020, 19:22:34	31591176	1_1	GNU c++ 11 4.9	OK	-	67ms	956.00Kb	-	-	отчёт

Контакты преподавателей:

Крымов Алексей, **СИНИЕ** (WEB-11/ML-11), krymov@corp.mail.ru, TG @alkrymov
Корепанов Дмитрий, **ЖЕЛТЫЕ** (WEB-12/ML-12), dmitrykorepanov@gmail.com, TG @DKorepanov

Глушенков Дмитрий, **ЗЕЛЕНЫЕ** (WEB-13/ML-13), d.glushenkov@corp.mail.ru, TG @gloonix

Мы гарантируем, что обратная связь по письму будет в течение 7 календарных дней в почту или по оперативному контакту из письма.

Правила сдачи РК

1. РК представляет собой конкурс с задачами, которые надо решить онлайн. Конкурс стартует в определенное время и у вас будет 1-1.5 часа (в зависимости от РК) на решение задач. По истечению времени конкурс закрывается.
2. Студент присылает ссылки на задачи в почту преподавателю.
3. Если задача выполнена в соответствии с условием и все тесты проходят, то задача засчитывается.

Минимальные правила кодирования.

1. Только C++.
2. Наличие среды разработки с поддержкой пошаговой отладки.
3. Наличие комментариев, объясняющих нетривиальные моменты в решении.
4. Необходимо придерживаться одного стиля написания кода.
Возможные варианты:
Google: <http://google.github.io/styleguide/cppguide.html>, автоматическая проверка: <http://cpplint.appspot.com/>
ABBY: <https://drive.google.com/open?id=0B6oYnrwl0UcLTRDVHNxVGxJeUU>
GNU: <https://gcc.gnu.org/codingconventions.html>
Поиск@Mail.ru: https://github.com/akrymov/technopark_common/blob/master/cppguide.md
5. Каждая сдаваемая задача предварительно должна быть сдана в тестирующую систему.
6. В шапке сдаваемого сср должно находиться закомментированное условие задачи.
7. Для проверки входных данных и работы встроенных функций рекомендуется использовать `assert(...)`.
8. Массивы неконстантного размера, а также большого константного размера необходимо выделять в динамической памяти.
9. На каждый вызов аллокации памяти должен быть вызов освобождения памяти, `new -> delete`. Начиная со второго модуля штраф "-1" за любую утечку памяти в сдаваемом и показываемом коде.
10. Вызов `delete` должен быть в той же логической области видимости что и вызов `new` (либо в той же функции, либо в том же классе).
11. Код решения должен быть грамотно разбит на функции: в `main` должен находиться ввод данных, вызов функции, которая решает задачу, вывод результата. Ввод/вывод производится только в `main`, решение - только в отдельном наборе функций и/или классов.
12. Каждой структуре данных должен соответствовать класс с продуманным стандартным публичным интерфейсом.
13. Все переменные и функции должны иметь осмысленное имя. Однобуквенные имена разрешаются только для счётчиков цикла и для переменных, заданных в условии задачи.
14. Переменные должны объявляться по месту использования.

15. Все переменные базового типа должны при объявлении инициализироваться, классы и структуры должны иметь конструктор.
16. Необходимо соблюдать правило трёх: если у класса есть деструктор, конструктор копирования и/или оператор присваивания, то должны быть все три.
17. Глобальными переменными пользоваться нельзя.
18. Рекурсией произвольной глубины пользоваться нельзя. Ей можно пользоваться, только если вы уверены, что глубина не превысит 1000.
19. Для каждой задачи необходимо определить скорость работы и потребляемую память.
20. Реализованное решение должно быть максимально эффективным.
21. **Любое дублирование чужого кода штрафуются “-5” и незачёт по задаче. Запрещено списывать и давать списывать. За это -5 и незачёт по задаче получают и списывающий, и дающий списывать.**