



Это первый хакатон из цикла мероприятий, посвященных подготовке к финалу Олимпиады НТИ. Он является логическим продолжением серии задач второго отборочного этапа и приближается к формату и содержанию командной задачи заключительного тура.

Задача, которую участники будут решать на хакатоне, состоит в определении характера шума в виртуальном канале связи и передаче в этом канале изображения без искажений.

Хакатон направлен на развитие следующих навыков и знаний у участников:

- работа в стрессовом и постоянно меняющемся окружении
- нахождение закономерностей в условиях неполноты информации
- работа в команде и разделение обязанностей

Ориентировочная длительность мероприятия: 5-6 часов

Минимальное и максимальное количество участников: 5 человек в команде

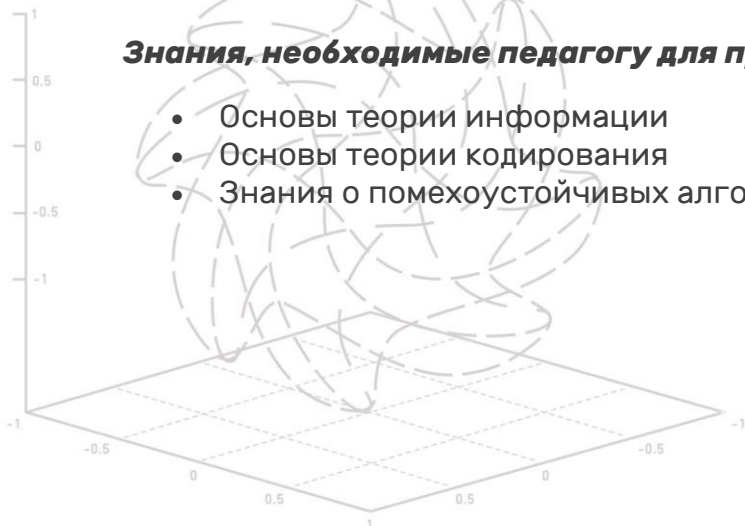
Список используемого оборудования:

1. Компьютеры с предустановленным ПО
 - a. Компиляторы языков (C++11, Java, Python)
 - b. IDE для вышеупомянутых языков (Visual Studio, PyCharm, IDEA, Eclipse)
 - c. библиотеки для работы с изображениями формата PNG (например: PIL для Python, LodePNG для C++, ImageIO для Java)
2. Локальная общая точка доступа
3. Сервер с установленным ПО для проведения хакатона, подключенный к локальной общей точке доступа.

Хакатон программный, для его проведения не требуется расходных материалов и оборудования кроме компьютеров с предустановленным ПО и файлов заданного формата и содержания.

Знания, необходимые педагогу для проведения мероприятия:

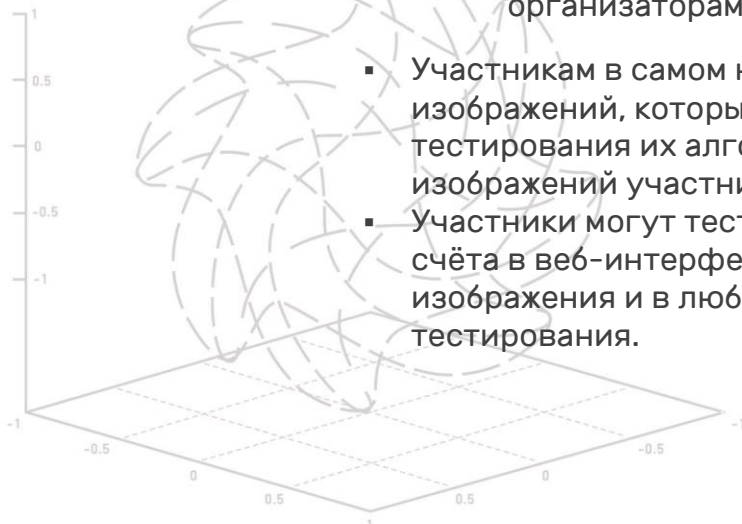
- Основы теории информации
- Основы теории кодирования
- Знания о помехоустойчивых алгоритмах





Регламент:

- Что проводящий должен упомянуть/объяснить во вступительном слове?
 - Объяснить порядок проведения хакатона, регламент и критерии оценки
 - Упомянуть, что на хакатоне природа шума может быть самой разной
 - Упомянуть про среду исполнения программ (Linux).
- Каков регламент и расписание?
 - Регламент:
 - **Участников необходимо предупредить за несколько дней до начала проведения о нескольких особенностях хакатона:**
 - Участники в ходе проведения хакатона могут (и должны) разрабатывать программу с двумя функциями: кодера и декодера. Эта программа будет исполняться в среде CentOS или Ubuntu (Linux).
 - Команды могут принести заранее подготовленные программы и материалы. Могут использовать собственные ноутбуки и другое оборудование. Подготовка к хакатону предпочтительна.
 - Задачи хакатона связаны с работой с изображениями (а именно: с PNG форматом), поэтому желательно иметь опыт работы (или попрактиковаться в работе) с PNG-изображениями.
 - Пользоваться интернетом во время проведения хакатона разрешено. Тем не менее, наличие интернета не обязательно для проведения и он может быть не предоставлен организаторами.
 - Дать необходимую информацию о доступности программных языков и IDE на предоставляемых организаторами ноутбуках.
 - Участникам в самом начале хакатона доступны несколько изображений, которые будут использоваться для тестирования их алгоритмов, однако некоторая часть изображений участникам будет недоступна.
 - Участники могут тестировать свою программу без изменения счёта в веб-интерфейсе с использованием выбранного изображения и в любой момент могут завершить текущий этап тестирования.

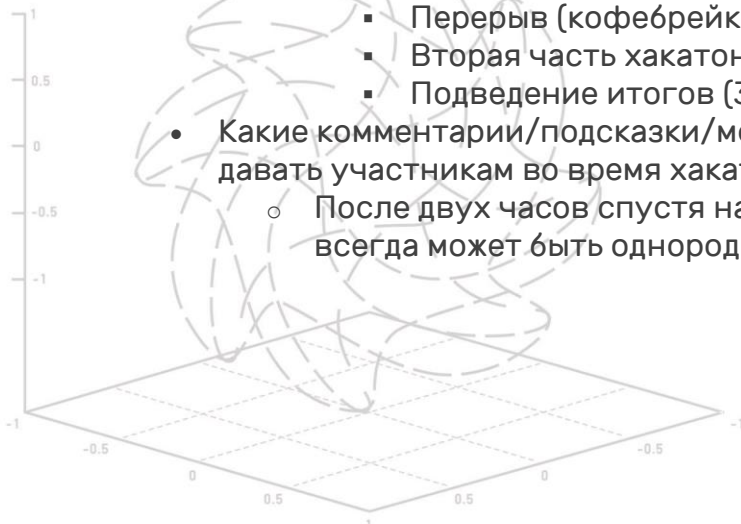




- При завершении этапа тестирования оценивается работа кодера и декодера по изображениям, которые доступны команде и некоторой доле тех, которые ей недоступны.
- После завершения командой этапа тестирования, или каждые полтора часа, доля недоступных изображений, которая была использована для текущего этапа, становится доступной.
- Через каждые полтора часа, во время проведения хакатона, показывается счёт за текущий этап, который вычисляется так: $100 \cdot \frac{\text{Общее сгенерированное количество ошибок} - \text{Число пропущенных ошибок}}{\text{Общее сгенерированное количество ошибок}}$ (точность работы алгоритмов команды). В это же время показывается общая точность команды (среднее) и общее штрафное (бонусное) время.
- За весь хакатон характер шума меняется 2 раза:
 - Начало хакатона: шум с вероятностью 100% меняет входной бит с “0” на “1” и с “1” на “0” (инвертирует) в predetermined местах. (вектор ошибок статичен для всех изображений)
 - Спустя 1.5 часа: шум с вероятностью p_1 (задаётся проводящим хакатона по тому, насколько хорошо справляются участники с первым источником шума) меняет в случайных местах биты.
 - Спустя 1.5 часа: шум с вероятностью p_2 и вероятностью $f(N)$, где $f(N): N \rightarrow [0,1]$ - функция плотности распределения ($f(x) - \text{tri}(0,0,2 / q)$) образует цепочки ошибок длиной до N бит. $f(x)$ - прямая, которая заканчивается при $f(x)=0$ и начинается в точке $(0, q)$, $q \in [0, 1]$ (образует серии ошибок).
 - Важно: количество ошибок у каждой команды одинаково на каждом этапе, а цепочки ошибок одинаковой длины!

○ Расписание:

- Вступительное слово и вводная часть (20 минут)
- Первая часть хакатона (3 часа)
- Перерыв (кофебрейк на 30 минут)
- Вторая часть хакатона (1.5 часа)
- Подведение итогов (30 минут)
- Какие комментарии/подсказки/методические указания может прийти давать участникам во время хакатона
 - После двух часов, спустя начала хакатона, можно указать, что шум не всегда может быть однородным и постоянным во времени





Суть задачи на хакатон:

Командам необходимо правильно передавать изображения в канале с помехами. Команды пишут программы, которые работают в режимах кодера и декодера. Эти программы тестируются на сериях изображений, некоторая часть которых всегда доступна, и эта часть постоянно растёт.

Тестирование проводится на удалённой машине.

- Критерии решения/победы/набора баллов:
 - Каждая ошибка декодирования на этапе тестирования считается за 1 штрафной балл.
 - Если команда закончила этап раньше 1.5 часов, количество штрафных баллов за этот этап уменьшается на: $20 \cdot \text{количество оставшихся секунд} / 5400$
 - Если команда закончила этап позже 1.5 часов, количество штрафных баллов за этот этап увеличивается на: $100 \cdot \text{количество штрафных секунд} / 5400$. Общее количество штрафных баллов, тем не менее, не может превысить возможное количество штрафных баллов.
- Что про нее должен объяснить педагог?
 - Работу штрафного и бонусного времени объяснять в деталях не нужно.
 - Нужно также пояснить, что, если закончить текущий этап раньше, можно проходить следующий этап дольше.

Формат подведения итогов:

- Критерии решения/победы/набора баллов
 - Критерий победы: наименьшее число штрафных баллов. Второй критерий - наименьшее штрафное время или наибольшее - бонусное.
- Что необходимо обсудить по итогам проведения мероприятия? какие рекомендации выдать участникам для дальнейшей подготовки?
 - Список литературы по методам кодирования и теории информации.

