

**АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОРГАНИЗАЦИЯ  
ДОПОЛНИТЕЛЬНОГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ  
«ИНСТИТУТ РАЗВИТИЯ КОМПЕТЕНЦИЙ»**

УТВЕРЖДАЮ:

Директор АНО ДПО

«Институт развития компетенций»

Шелихова М.М.

28 апреля 2023 г.



**Сборник заданий**

**Регионального командного инженерного первенства «КУБИТВА»**

ХМАО-Югра, 2023 г.

# I. ЗАДАНИЯ ОТБОРОЧНОГО ЭТАПА

## РЕГИОНАЛЬНОГО КОМАНДНОГО ИНЖЕНЕРНОГО ПЕРВЕНСТВА «КУБИТВА»

### Назначение документа:

Задания предназначены для проведения внутреннего конкурсного отбора участников Проектной школы среди обучающихся Центров цифрового образования детей «IT-куб» Ханты-Мансийского автономного округа – Югры.

### Формат проведения отбора:

Индивидуальное выполнение заданий в дистанционном или очном формате (по выбору «IT-куба») без использования внешних источников и систем искусственного интеллекта.

**Рекомендуемое время выполнения:** 90–120 минут.

**Проходной балл:** 60 баллов из 100.

### 1.1. НАПРАВЛЕНИЕ: «ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ»

#### ОПИСАНИЕ ЗАДАНИЯ

#### Блок 1. Мотивация и эрудиция (15 баллов)

1. Почему Вы хотите заниматься информационной безопасностью? (3–5 предложений)
2. Приведите пример реальной кибератаки из новостей за последние 2–3 года (укажите год, цель атаки, последствия).
3. Выберите один из терминов: **фишинг**, **DDoS-атака**, **троян**, **кейлоггер**. Дайте определение и объясните, чем он опасен для обычного пользователя.

#### Блок 2. Логика и алгоритмы (30 баллов)

1. Расшифруйте сообщение, закодированное с помощью XOR с ключом 42 (в десятичной системе). Известно, что исходное сообщение — английское слово. Закодированная последовательность (в шестнадцатеричной системе): 4B 4B 4C 57 46 41 47.

*Подсказка: XOR — побитовое исключающее ИЛИ. Символы в ASCII.*

2. Что такое «атака перебором» (**brute force**)? В каких случаях она эффективна, а в каких — нет? Приведите пример защиты от перебора.

3. Проанализируйте фрагмент кода:

text

```
String query = "SELECT * FROM users WHERE login = '" + request.getParameter("login") + "'";
```

Какая уязвимость здесь присутствует? Чем она опасна? Как её исправить?

### Блок 3. Прикладная задача (30 баллов)

Вы — системный администратор небольшой компании. Ваш коллега получил письмо по электронной почте: отправитель — [support@bank-online.com](mailto:support@bank-online.com) (домен похож на реальный банк, но отличается одной буквой), в письме говорится, что его счёт заблокирован, и требуется перейти по ссылке <http://bank-online-verify.com> и ввести логин и пароль.

#### Задания:

1. Определите тип атаки (название).
2. Перечислите не менее трёх признаков, по которым можно распознать это письмо как вредоносное.
3. Опишите, как должен поступить коллега (какие действия предпринять, чтобы не стать жертвой).

### Блок 4. Рефлексия (25 баллов)

1. Сталкивались ли Вы лично с попыткой взлома Ваших аккаунтов (социальные сети, игры, почта)? Если да — опишите, что произошло и как Вы действовали.
2. Какие меры защиты Вы используете для своих аккаунтов (пароли, двухфакторная аутентификация, антивирус и т.д.)?
3. Что для Вас было самым трудным в этом задании? Как Вы искали решение?

## 1.2. НАПРАВЛЕНИЕ: «РАЗРАБОТКА КОМПЬЮТЕРНЫХ ИГР»

### ОПИСАНИЕ ЗАДАНИЯ

#### Блок 1. Мотивация и понимание геймдева (15 баллов)

1. Почему Вы хотите разрабатывать игры? Что Вас привлекает в этой сфере?
2. Назовите игру, которая, по Вашему мнению, имеет **идеальный онбординг** (обучение игрока). Поясните, почему обучение в этой игре сделано хорошо.
3. Что такое «**игровой баланс**»? Приведите пример несбалансированной игры (что именно было не так).

#### Блок 2. Логика и программирование (30 баллов)

1. Напишите псевдокод для следующей механики: в игре есть сундук, который можно открыть, только если у игрока есть **золотой ключ**. При открытии сундука из него выпадает случайный предмет (из трёх возможных: меч, зелье, монеты). Ключ при этом расходуется.
2. Что такое **коллизия** в игровом движке? Зачем она нужна? Какие бывают формы коллизий (хотя бы две)?
3. Дан фрагмент кода на C# (Unity):

```

csharp
void Update()
{
    if (Input.GetKeyDown(KeyCode.Space))
    {
        transform.Translate(Vector3.up * 5);
    }
}

```

Что делает этот код? Какая проблема может возникнуть при его использовании? Как её исправить?

### Блок 3. Проектная задача (30 баллов)

Опишите концепцию **мобильной игры-головоломки** (жанр — «три в ряд» или «match-3»).

Используйте шаблон:

Элемент концепции	Ваш ответ
Название игры	
Сеттинг (тема, окружающий мир)	
Основная механика (что делает игрок)	
Цель игры	
Как игра становится сложнее	
Способ монетизации (реклама / донаты / платная загрузка)	

**Дополнительно:** нарисуйте пример игрового поля (схематично, можно от руки и сфотографировать).

### Блок 4. Рефлексия (25 баллов)

1. В каком игровом движке Вы пробовали что-то делать (Unity, Unreal, GameMaker, Scratch, конструкторы)?
2. Что было самым сложным в освоении программирования / разработки игр для Вас лично?
3. Какой игровой проект Вы хотели бы создать в будущем? Почему?

**МАКСИМАЛЬНЫЙ БАЛЛ: 100.**

## II. ЗАДАНИЯ ФИНАЛА

### РЕГИОНАЛЬНОГО КОМАНДНОГО ИНЖЕНЕРНОГО ПЕРВЕНСТВА «КУБИТВА»

#### ОБЩАЯ ИНСТРУКЦИЯ

Параметр	Значение
Формат проведения	очно, в рамках Проектной школы «Джуниор АйТи. Битва Кубов»
Время на выполнение	4 академических часа (3 часа — работа над кейсом, 1 час — подготовка защиты)
Время на защиту	до 15 минут (10 минут — выступление, 5 минут — вопросы жюри)
Формат работы	командный (от 2 до 5 человек от одного «ИТ-куба» внутри направления)
Доступ к ресурсам	разрешена официальная документация, запрещены готовые решения и ИИ

#### 2.1. НАПРАВЛЕНИЕ: «ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ»

##### 2.1.1. Описание формата финала

Командам предстоит выполнить задание по расследованию инцидента в небольшой компании. Предоставлены: логи веб-сервера, дампы сетевого трафика (упрощённый) и текстовое описание ситуации. Участники должны восстановить ход атаки, выявить уязвимости и предложить меры защиты.

##### 2.1.2. Комплексное задание (кейс)

###### Контекст

Компания «ВебСервис» разрабатывает интернет-магазин. В ночь с 14 на 15 марта 2023 года произошла утечка данных клиентов: в открытом доступе оказались логины, пароли (в открытом виде) и номера телефонов. Администратор компании предоставил фрагменты логов за 14.03.2023.

###### Лог веб-сервера (фрагмент)

log

```
10.0.0.5 -- [14/Mar/2023:10:00:15 +0000] "GET /index.html HTTP/1.1" 200
10.0.0.5 -- [14/Mar/2023:10:00:22 +0000] "GET /catalog HTTP/1.1" 200
198.51.100.7 -- [14/Mar/2023:10:15:10 +0000] "GET /login HTTP/1.1" 200
198.51.100.7 -- [14/Mar/2023:10:15:13 +0000] "POST /login HTTP/1.1" 302
198.51.100.7 -- [14/Mar/2023:10:15:15 +0000] "GET /admin/dashboard HTTP/1.1" 403
198.51.100.7 -- [14/Mar/2023:10:15:30 +0000] "GET /admin/dashboard?user=admin' OR '1'=1"
```

```
HTTP/1.1" 200
```

```
198.51.100.7 - - [14/Mar/2023:10:15:45 +0000] "GET /api/users/export HTTP/1.1" 200
```

```
10.0.0.5 - - [14/Mar/2023:10:20:00 +0000] "GET /index.html HTTP/1.1" 200
```

Дамп сетевого трафика (описание, вместо PCAP) — файл `traffic.txt`

text

```
14.03.2023 10:15:10.123 198.51.100.7:54321 -> 10.0.0.5:80 [TCP SYN]
```

```
14.03.2023 10:15:10.125 10.0.0.5:80 -> 198.51.100.7:54321 [TCP SYN-ACK]
```

```
14.03.2023 10:15:10.130 198.51.100.7 -> 10.0.0.5 HTTP GET /login
```

```
14.03.2023 10:15:13.200 198.51.100.7 -> 10.0.0.5 HTTP POST /login (data:
user=admin&pass=12345)
```

```
14.03.2023 10:15:13.250 10.0.0.5 -> 198.51.100.7 HTTP 302 Found (Location: /admin/dashboard)
```

```
14.03.2023 10:15:30.100 198.51.100.7 -> 10.0.0.5 HTTP GET /admin/dashboard?user=admin' OR
'1'=1
```

```
14.03.2023 10:15:30.200 10.0.0.5 -> 198.51.100.7 HTTP 200 OK (content: admin panel)
```

```
14.03.2023 10:15:45.100 198.51.100.7 -> 10.0.0.5 HTTP GET /api/users/export
```

```
14.03.2023 10:15:45.300 10.0.0.5 -> 198.51.100.7 HTTP 200 OK (content: user data in JSON)
```

Текстовый дамп памяти (упрощённый) — файл `memory.txt`

text

```
[PID: 2345] /usr/sbin/apache2
```

```
[2345] Environment: DB_PASSWORD=secret2023
```

```
[2345] Recent SQL queries:
```

```
10:15:13 SELECT * FROM users WHERE login='admin' AND pass='12345' -> 0 rows
```

```
10:15:30 SELECT * FROM users WHERE login='admin' OR '1'=1 -> 1 row (admin)
```

```
[2345] In-memory cache: user_list: ["alex", "maria", "ivan", "admin"]
```

### 2.1.3. Задания для команд

#### Этап 1. Анализ (30% оценки)

Используя предоставленные файлы:

1. Определите IP-адрес злоумышленника.
2. Какой тип атаки был использован для получения доступа к панели администратора?
3. Благодаря какой уязвимости это стало возможно?
4. Какие данные были похищены?

#### Этап 2. Восстановление цепочки атаки (40% оценки)

1. Опишите хронологию действий злоумышленника (что в какой момент он делал).
2. Какие ошибки допустила компания в настройке безопасности?
3. Какая учётная запись была скомпрометирована?

### Этап 3. Рекомендации по защите (30% оценки)

Подготовьте отчёт для руководства (не более 1 страницы):

1. Три главные уязвимости, которые необходимо устранить.
2. Конкретные меры по каждой уязвимости.
3. Как изменить политику паролей и хранения данных.

#### 2.1.4. Защита решения

- Презентация (5–7 слайдов) по трём этапам.
- Устное выступление (7–10 минут).
- Ответы на вопросы жюри.

#### 2.1.5. Критерии оценки

Критерий	Макс. балл	Пояснение
Полнота решения	25	все три этапа выполнены
Корректность	25	правильно определён IP, тип атаки, уязвимости, цепочка
Техническая сложность	20	глубина анализа, использование дополнительных признаков
Оригинальность	10	нестандартный подход, креативные рекомендации
Качество защиты	20	логичная презентация, уверенные ответы
<b>ИТОГО</b>	<b>100</b>	

## 2.2. НАПРАВЛЕНИЕ «РАЗРАБОТКА КОМПЬЮТЕРНЫХ ИГР»

### 2.2.1. Описание формата финала

Командам предстоит разработать игральный прототип **2D-платформера «Лесной собиратель»** на выбор: в Unity (2021 LTS) или в конструкторе Scratch (для менее подготовленных участников — решение принимает Организатор). Прототип должен демонстрировать ключевые механики и быть представленным жюри.

### 2.2.2. Техническое задание «Лесной собиратель»

**Жанр:** 2D-платформер

**Цель игры:** Управляя ёжиком, собрать 5 яблок, разбросанных по уровню, избегая встречи с волком. После сбора всех яблок открывается выход на следующий уровень.

**Обязательные механики:**

- Управление ёжиком (влево/вправо — стрелки, прыжок — пробел). Реализовано через физику (Rigidbody2D, коллайдеры).
- Сбор яблок: при касании яблоко исчезает, счёт увеличивается на 1.
- Препятствие — волк: при касании ёжика с волком игра возвращается к началу уровня, счёт не обнуляется, но теряется одна жизнь. Жизни: 3 (отображаются в виде сердечек в углу экрана).
- Отображение счётчика яблок (0/5).
- При сборе 5 яблок появляется «домик» (закрытая дверь).
- При касании домика появляется сообщение «You win!» и игра останавливается (или загружается новый уровень — по желанию).

#### **Дополнительные механики (бонус до +10 баллов):**

- Анимация ёжика (идёт, прыгает, собирает).
- Ловушка (яма), при падении в которую игрок теряет жизнь и возвращается на старт.
- Фоновая музыка и звук сбора яблок.

#### **Технические требования (для Unity):**

- Версия Unity: 2021 LTS (или более новая)
- Все ассеты могут быть взяты из бесплатных источников (Kenney.nl, Unity Asset Store) или созданы командой.
- Проект должен быть собран в исполняемый файл (.exe) и представлен вместе с исходниками.

#### **Технические требования (для Scratch):**

- Проект должен быть экспортирован в .sb3 или опубликован на сайте Scratch.
- Ассеты — встроенная библиотека или загруженные изображения.

#### **2.2.3. Защита решения**

- Демонстрация игры (вживую) — до 5 минут.
- Презентация (3–5 слайдов) — механики, распределение ролей, использованные ассеты.
- Вопросы жюри (3–5 минут).

#### **2.2.4. Критерии оценки**

Критерий	Макс. балл	Пояснение
Полнота реализации ТЗ	30	все обязательные механики работают
Корректность	20	нет критических багов (застревание, неправильные коллизии)
Качество игровых механик	15	плавное управление, баланс, играбельность

Критерий	Макс. балл	Пояснение
Визуальное оформление / UI	15	эстетика, читаемость счётчика и жизней
Качество защиты	20	живая демонстрация, аргументированные ответы
Бонус (доп. механики)	до +10	анимация, ловушка, звук
<b>ИТОГО</b>	<b>100 (+10)</b>	

### 2.2.5. Рекомендации для команд (выдаются на финале)

- Распределите роли: программист, художник, дизайнер уровней, тестировщик, презентатор.
- Проверьте коллизии: ёжик не должен проваливаться сквозь землю.
- Убедитесь, что счёт яблок обнуляется при потере жизни (по замыслу — не обнуляется).
- Соберите билд (если Unity) и проверьте его на другом компьютере.