



**АЛТАЙСКИЙ
ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ**

Инструкция по добавлению плагина на курс

Программирование Deertalk

Добавление элемента.

The screenshot shows the top navigation bar with links: Главная, Личный кабинет, Курсы, Этот курс, Преподавателям, and Оплатить онлайн. Below the navigation bar, the breadcrumb path is 'Мои курсы > Курс для примера'. The main content area has tabs: Курс, Настройки, Участники, Оценки, Отчеты, and Дополнительно. The 'Общее' (General) section is expanded, showing 'Объявления' (Announcements) and 'Регламент деловых коммуникаций педагогических работников и обучающихся' (Code of conduct for pedagogical workers and students). On the right, a settings menu is open, with 'Режим редактирования' (Edit mode) highlighted by a blue arrow.

Для того что бы добавить плагин перейдем в режим редактирования курса.

The screenshot shows a button with a plus sign and the text 'Добавить элемент или ресурс' (Add element or resource). A blue arrow points to the button.

Добавить темы

The screenshot shows a grid of plugin options. The 'Программирование DeepTalk' (DeepTalk Programming) plugin is highlighted with a blue arrow. The grid includes the following options:

- Лекция (ДТ) (Lecture (DT))
- Медиаколлекция (Media collection)
- Опрос (Survey)
- Папка (Folder)
- Посещаемость (Attendance)
- Программирование DeepTalk (DeepTalk Programming)
- Текст и медиа (Text and media)
- Тест (Test)
- Файл (File)

Переходим к нужному блоку и нажимаем добавить элемент или ресурс. Затем выбираем плагин Программирование DeepTalk.

Настройка элемента.

Добавление: Программирование DeepTalk

Основные настройки

Название !

Язык программирования !

Тип задания !

Режим тренировки

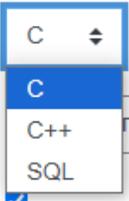
Перед Вами интерфейс настройки элемента.

В первую очередь мы выбираем Язык программирования.

Язык программирования !

Тип задания !

Режим тренировки

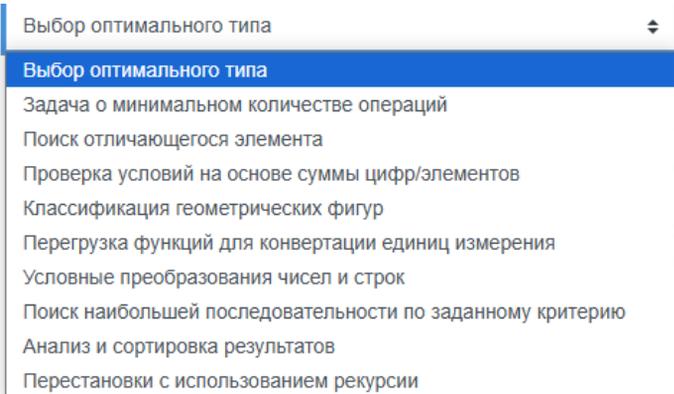


Далее необходимо выбрать тип задания, типы заданий меняются в зависимости от выбора языка программирования.

Тип задания !

Режим тренировки

Описание задания



Режим тренировки, если галочка стоит, то оценки нет за данный элемент. Если убрать галочку, то данный элемент будет автоматически оцениваться и идти в оценку студента.

Режим тренировки



Остается только дать название элементу и нажать в конце, сохранить и показать.

Сохранить и вернуться к курсу

Сохранить и показать

Отмена

Работа с элементом.

Классификация геометрических фигур на C

Классифицировать геометрические фигуры в зависимости от их размера.

Язык программирования C

Тип задания Классификация геометрических фигур

Вид задания Тренировка

Описание Классифицировать геометрические фигуры в зависимости от их размера.

Дата создания	Статус	Режим	Отчет	
12.03.2025, 14:20:17	Отправлен результат	Базовый	Просмотреть	⊞
12.03.2025, 14:20:27	Отправлен результат	Базовый	Просмотреть	⊞
12.03.2025, 14:25:01	Отправлен результат	Продвинутый	Просмотреть	⊞

[Получить задание](#)

Каждому студенту необходимо нажать кнопку получить задание.

Задание формируется для любого пользователя свое. При ответе он видит задания и поле для написания кода.

Предоставление ответа на задание возможно один раз.

1 Задание

В архитектурном бюро 'Геометрия будущего' необходимо разработать программу на языке C для анализа формы нового выставочного павильона в форме пятиугольника. Даны стороны павильона: 48.5, 65.0, 89.5, 111.0 и 75.5 метров, и внутренние углы: 108.5, 115.0, 98.5, 122.0 и 96.0 градусов.

Напишите программу, которая: 1. Определит тип пятиугольника (правильный или неправильный). 2. Вычислит периметр павильона. 3. Проверит, является ли пятиугольник выпуклым (сумма внутренних углов должна быть равна 540 градусам).

Требования к реализации: 1. Создайте функцию checkPolygonType(), которая принимает массивы сторон и углов и возвращает тип пятиугольника. 2. Создайте функцию calculatePerimeter(), которая вычисляет периметр. 3. Создайте функцию isConvex(), которая проверяет выпуклость пятиугольника.

Формат вывода: Тип пятиугольника: [правильный/неправильный] Периметр: [значение] метров Выпуклость: [да/нет]

Примечание: При проверке на правильность многоугольника учитывайте погрешность в 0.1 единицы измерения.

```
}
int main() {
    // Входные данные: длины сторон и углы
    double sides[] = {48.5, 65.0, 89.5, 111.0, 75.5};
    double angles[] = {108.5, 115.0, 98.5, 122.0, 96.0};

    // Определение типа пятиугольника
    const char* type = checkPolygonType(sides, angles);

    // Вычисление периметра
    double perimeter = calculatePerimeter(sides);

    // Проверка выпуклости
    int convex = isConvex(angles);

    // Вывод результатов
    printf("Тип пятиугольника: %s\n", type);
    printf("Периметр: %.2f метров\n", perimeter);
    printf("Выпуклость: %s\n", convex ? "да" : "нет");

    return 0;
}
```

[Назад](#)

[Отправить ответ](#)



Работа с элементом.

Отчет по заданию

Рекомендации

#1 Добавить комментарии к каждой функции, чтобы пояснить их назначение и входные параметры. Это улучшит читаемость и понимание кода. Пример:

```
// Функция проверяет тип пятиугольника (правильный или неправильный)
const char* checkPolygonType(double sides[], double angles[]);
```

#2 Использовать именованные константы вместо магических чисел, таких как 5 (количество сторон) и 540. Это сделает код более понятным и облегчит его поддержку. Пример:

```
#define NUM_SIDES 5
#define TOTAL_ANGLE_SUM 540.0
```

#3 Добавить проверку входных данных на корректность, например, чтобы длины сторон были положительными и углы находились в диапазоне от 0 до 180 градусов. Это повысит надежность программы. Пример:

```
for (int i = 0; i < 5; i++) {
    if (sides[i] <= 0 || angles[i] <= 0 || angles[i] >= 180) {
        printf("Ошибка: некорректные входные данные.\n");
    }
    return -1;
}
}
```

После того, как студент отправляет ответ, он получает отчет по заданию и рекомендации.

Если вид задания: Оценивание, то студент может получить задание один раз и после его отправки получит оценку зачёт или незачет.

Язык программирования C

Тип задания Проверка условий на основе суммы цифр/элементов

Вид задания Оценивание

Описание 32141241212

Дата создания	Статус	Режим	Отчет	Оценка
12.03.2025, 14:27:41	Отправлен результат	Базовый	Просмотреть	незачет



Работа с элементом.

Преподаватель может посмотреть отчёт по каждому студенту.

Язык программирования C
Тип задания Проверка условий на основе суммы цифр/элементов
Вид задания Оценивание
Описание 32141241212

Дата создания	Статус	Режим	Отчет	Оценка	
11.03.2025, 11:16:03	Отправлен результат	Базовый	Просмотреть	зачет	↻

[Отчет](#)

В этом разделе, можно увидеть всех пользователей, что проходили данное задание, их оценки, отчёт и кол-во попыток.

Пользователь	Количество попыток
Шмаченко 1 Это режим тренировки.	4

Дата создания	Статус	Режим	Отчет
12.03.2025, 14:20:17	Отправлен результат	Базовый	Просмотреть
12.03.2025, 14:20:27	Отправлен результат	Базовый	Просмотреть
12.03.2025, 14:25:01	Отправлен результат	Продвинутый	Просмотреть
12.03.2025, 14:32:52	Отправлен результат	Продвинутый	Просмотреть

Пользователь	Количество попыток
Шмаченко Александр Это режим оценивания.	1

Дата создания	Статус	Режим	Отчет	Оценка
11.03.2025, 11:16:03	Отправлен результат	Базовый	Просмотреть	зачет

Пользователь	Количество попыток
Шмаченко 1	1

Дата создания	Статус	Режим	Отчет	Оценка
12.03.2025, 14:27:41	Отправлен результат	Базовый	Просмотреть	незачет

[Назад](#)