

Подписано электронной подписью:  
Вержицкий Данил Григорьевич  
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»  
Дата и время: 2024-02-21 00:00:00  
471086fad29a3b30e244c728abc3661ab35c9d50210dcf0e75e03a5b6fdf6436

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение высшего образования  
«КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
Кузбасский гуманитарно-педагогический институт

---

Факультет физики, математики и экономики

УТВЕРЖДАЮ  
Декан ФИМЭ  
А.В.Фомина  
«10» февраля 2023 г.

**Рабочая программа дисциплины**  
**К.М.06.03 Выравнивающий курс математики и программирования**  
*Код, название дисциплины /модуля*

Направление подготовки / *специальность*  
**09.03.03 Прикладная информатика**

Направленность (профиль) программы / специализация  
**Прикладная информатика в образовании**

Программа бакалавриата  
Квалификация выпускника  
*бакалавр*

Форма обучения  
*заочная*

Год набора 2023

Новокузнецк 2023

**Лист внесения изменений**

в РПД \_\_\_\_\_

*(код по учебному плану, название дисциплины)*

**Сведения об утверждении:**

утверждена Ученым советом факультета \_\_\_\_\_  
(протокол Ученого совета факультета № \_\_\_\_\_ от \_\_\_\_\_.20\_\_ г.)

для ОПОП 20\_\_ год набора \_\_\_\_\_ на 20\_\_ / 20\_\_ учебный год  
по направлению подготовки / специальности \_\_\_\_\_  
*(код и название направления подготовки / специальности)*

направленность (профиль) программы / специализация \_\_\_\_\_

Одобрена на заседании методической комиссии факультета \_\_\_\_\_  
протокол методической комиссии факультета № \_\_ от \_\_.\_\_.20\_\_ г.)

Одобрена на заседании обеспечивающей кафедры \_\_\_\_\_  
протокол № \_\_ от \_\_.\_\_.20\_\_ г. \_\_\_\_\_ (Ф. И.О. зав. кафедрой)

## Оглавление

1	Цель дисциплины. ....	4
1.1	Формируемые компетенции .....	<b>Ошибка! Закладка не определена.</b>
1.2	Индикаторы достижения компетенций .....	4
1.3	Знания, умения, навыки (ЗУВ) по дисциплине .....	<b>Ошибка! Закладка не определена.</b>
2	Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации. ....	5
3	Учебно-тематический план и содержание дисциплины. ....	6
3.1	Учебно-тематический план .....	6
3.2	Содержание занятий по видам учебной работы .....	<b>Ошибка! Закладка не определена.</b>
4	Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации. ....	6
5	Материально-техническое, программное и учебно-методическое обеспечение дисциплины. ....	7
5.1	Учебная литература .....	7
5.2	Материально-техническое и программное обеспечение дисциплины. ....	7
5.3	Современные профессиональные базы данных и информационные справочные системы. ....	8
6	Иные сведения и (или) материалы. ....	8
6.1	Примерные темы письменных учебных работ.....	8
6.2	Примерные вопросы и задания / задачи для промежуточной аттестации	10

### 1 Цель дисциплины.

В результате освоения дисциплины у обучающегося должны быть сформированы компетенции основной профессиональной образовательной программы бакалавриата (далее - ОПОП):  
ОПК-1;ОПК-2

### Формируемые компетенции, индикаторы достижения компетенций, знания, умения, навыки

Таблица 1 – Индикаторы достижения компетенций, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции по ОПОП	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
ОПК-1 Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности	ОПК 1.2 Применяет методы высшей и дискретной математики для моделирования прикладных и информационных процессов ОПК 1.3 Применяет методы теории вероятности и математической статистики для моделирования прикладных и информационных процессов	Знать : - основные принципы математической логики, теории множеств и теории графов. - формулы алгебры высказываний. - методы алгебраических преобразований. - основы языка и алгебры предикатов. - основные принципы теории множеств. - основные понятия комбинаторики; - основы теории вероятностей и математической статистики; - основные понятия теории графов. Уметь : - применять логические операции, формулы логики, законы алгебры логики, алгоритмы в графах. - формулировать задачи логического характера и применять средства математической логики для их решения. - применять стандартные методы и модели к решению вероятностных и статистических задач; - пользоваться расчетными формулами, таблицами, графиками при решении статистических задач; - применять современные пакеты прикладных программ многомерного статистического анализа
ОПК-2 Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной	ОПК 2.1 Осуществляет и обосновывает выбор современных информационных технологий и программных средств	Знать : - понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. - эволюцию языков программирования, их классификацию, понятие системы программирования.

Код и название компетенции	Индикаторы достижения компетенции по ОПОП	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
деятельности		<ul style="list-style-type: none"> <li>- основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.</li> <li>- подпрограммы, составление библиотек подпрограмм</li> <li>- объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка.</li> </ul> <p>Уметь</p> <ul style="list-style-type: none"> <li>: - разрабатывать алгоритмы для конкретных задач.</li> <li>- использовать программы для графического отображения алгоритмов.</li> <li>- определять сложность работы алгоритмов.</li> <li>- работать в среде программирования.</li> <li>- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.</li> <li>- оформлять код программы в соответствии со стандартом кодирования.</li> <li>- выполнять проверку, отладку кода программы</li> </ul>

## 2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации.

Таблица 2 – Объём и трудоёмкость дисциплины по видам учебных занятий

Общая трудоёмкость и виды учебной работы по дисциплине, проводимые в разных формах	Объём часов по формам обучения		
	ОФО	ОЗФО	ЗФО <sup>1</sup>
1 Общая трудоёмкость дисциплины			72
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)			
Аудиторная работа (всего):			2
в том числе:			
лекции			
практические занятия, семинары			
практикумы			2
лабораторные работы			
Внеаудиторная работа (всего):			66
в том числе, индивидуальная работа обучающихся с преподавателем			

<sup>1</sup> Оставить формы, в которых реализуется ОПОП

подготовка курсовой работы (проекта) /контактная работа <sup>2</sup>			
групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем)			
творческая работа (эссе)			
3 Самостоятельная работа обучающихся (всего)			66
4 Промежуточная аттестация обучающегося - зачет			Зачет 1 2 з.е.

### 3. Учебно-тематический план и содержание дисциплины.

#### 3.1 Учебно-тематический план

Таблица 3 - Учебно-тематический план очной / заочной формы обучения

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость (всего час.)	Трудоемкость занятий (час.)									Формы текущ. контроля и промежуточной аттестации
			ОФО			ОЗФО			ЗФО			
			Аудиторн. занятия		СРС	Аудиторн. занятия		СРС	Аудиторн. занятия		СРС	
			лекц.	практ		лекц.	практ		лекц.	практ		
<b>Семестр 1</b>												
1.	1. Дискретная математика с элементами математической логики											
1	1.1 Введение в теорию множеств.	6									6	
2	1.2 Классическая логика высказываний	6									6	
4	1.4 Введение в теорию графов	6									6	
2.	2. Теория вероятностей и математическая статистика											
5	2.1 Теория вероятностей	6									6	
6	2.2 Математическая статистика	6									6	
3.	3. Основы алгоритмизации и программирования											
7	3.1 Основы алгоритмизации	6									6	
8	3.2 Основы программирования	6									6	
9	3.3 Основы программирования на языке Python	6									6	
10	3.4 Объектно-ориентированное программирование на языке Python	8								2	6	
18	Промежуточная аттестация - зачет	4										зачет
ИТОГО по семестру 1		72								2	66	
Всего по учебному плану:		72								2	66	

#### 4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.

Для положительной оценки по результатам освоения дисциплины обучающемуся необходимо выполнить все установленные виды учебной работы. Оценка результатов работы обучающегося в баллах (по видам) приведена в таблице 4.

Таблица 4 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по видам (БРС)

<sup>2</sup> Часы, выделенные в УП на курсовое проектирование в контактной форме (3 часа)

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы за освоение дисциплины (мин.-макс.)
Текущая учебная работа в семестре (Посещение занятий по расписанию и выполнение заданий)	<b>80</b>	Лабораторные работы (отчет о выполнении лабораторной работы) (1 работа).	5 баллов - посещение 1 пр. занятия и выполнение задания на 51-65% 10 баллов – посещение 1 пр. занятия и выполнение задания на 85,1-100%, самостоятельность и существенный вклад на занятии в работу группы, др.	0-10
		Индивидуальное задание (по теме на выбор) (2 работы)	20 баллов (выполнены минимально достаточные требования) 35 баллов (выполнены все требования)	0-70
<b>Итого по текущей работе в семестре</b>				<b>0-80</b>
Промежуточная аттестация (зачет)	<b>20</b>	Тест.	6 баллов (выполнено 70% заданий и более) 12 баллов (выполнено 100% заданий )	0-10
		Решение задачи 1.	6 баллов - 12 баллов	0-5
		Решение задачи 2.	8 баллов - 16 баллов	0-5
<b>Итого по промежуточной аттестации (зачету) по приведенной шкале (20 б.)</b>				<b>0-20</b>
<b>Суммарная оценка по дисциплине/ Сумма баллов текущей и промежуточной аттестации</b>				<b>0 – 100 б.</b>

Обучающемуся по ЗФО задание на самостоятельную работу и контрольную работу выдается на установочной сессии.

## **5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины.**

### **5.1 Учебная литература**

#### **Основная учебная литература**

1. Бородин А. Н. Элементарный курс теории вероятностей и математической статистики : учебное пособие для вузов / Бородин А. Н. - 9-е изд., стер. - Санкт-Петербург : Лань, 2023. - 256 с. - Рекомендовано Министерством общего и профессионального образования РФ в качестве учебного пособия для студентов вузов, обучающихся по нематематическим специальностям. - ISBN 978-5-507-47132-4.
2. Чернышев С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. - Москва : Юрайт, 2023. - 286 с. - (Высшее образование). - URL: <https://urait.ru/bcode/519949> (дата обращения: 16.06.2023). - ISBN 978-5-534-14350-8.

#### **Дополнительная учебная литература**

1. Бирюкова Л. Г.. Теория вероятностей и математическая статистика : Учебное пособие / Российский экономический университет им. Г.В. Плеханова; Российский экономический университет им. Г.В. Плеханова. - 2. - Москва : ООО "Научно-издательский центр ИНФРА-М", 2021. - 289 с. - ISBN 978-5-16-011793-5. - ISBN 978-5-16-101044-0.
2. Буйначев С. К. Основы программирования на языке Python : учебное пособие / С. К. Буйначев, Н. Ю. Боклаг. – Екатеринбург : Изд-во Урал. ун-та, 2014. – 91, [1] с. ISBN 978-5-7996-1198-9
3. Гисин В. Б. Дискретная математика : учебник и практикум для вузов / В. Б. Гисин. - 2-е изд. - Москва : Юрайт, 2023. - 468 с. - (Высшее образование). - URL: <https://urait.ru/bcode/531659> (дата обращения: 01.06.2023). - ISBN 978-5-534-16763-4.

## 5.2 Материально-техническое и программное обеспечение дисциплины.

Учебные занятия по дисциплине проводятся в учебных аудиториях КГПИ КемГУ:

<p><b>508 Компьютерный класс</b> Учебная аудитория для проведения занятий лекционного типа, занятий лабораторного типа, для групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации (мультимедийная)</p> <p><b>Специализированная (учебная) мебель:</b> доска меловая, кафедра, столы, стулья,</p> <p><b>Оборудование для презентации учебного материала:</b> компьютер преподавателя, проектор, экран, 18 компьютеров</p> <p><b>Лабораторное оборудование:</b> стационарное – компьютеры для обучающихся (18 шт.).</p> <p><b>Используемое программное обеспечение:</b> MS Windows (Microsoft Imagine Premium 3 year по лицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Scilab (свободно распространяемое ПО), UML-диаграммы (бесплатная версия), Консультант Плюс (отечественное ПО, договор об инфо поддержке 1.04.2007), Python3 (свободно распространяемое ПО),</p> <p><b>Интернет с обеспечением доступа в ЭИОС</b></p>	654079, Кемеровская область, г. Новокузнецк, пр-кт Metallургов, д. 19
---	---

## 5.3 Современные профессиональные базы данных и информационные справочные системы.

### Перечень СПБД и ИСС по дисциплине

1. Федеральный портал «Российское образование» - <http://www.edu.ru>. Доступ свободный
2. Информационная система «Единое окно доступа к образовательным ресурсам» - <http://www.window.edu.ru>.
3. Федеральный центр информационно-образовательных ресурсов - <http://fcior.edu.ru>. Доступ свободный.
4. Федеральный портал "Информационно-коммуникационные технологии в образовании" - <http://www.ict.edu.ru/>.
5. Сайт Министерства образования и науки РФ. - Режим доступа: <http://www.mon.gov.ru>. Доступ свободный.
6. Единая коллекция цифровых образовательных ресурсов.- Режим доступа: <http://school-collection.edu.ru/>
7. Единое окно доступа к образовательным ресурсам. Раздел Образование в области техники и технологий – [http://window.edu.ru/?p\\_rubr=2.2.75](http://window.edu.ru/?p_rubr=2.2.75)

### 6 Иные сведения и (или) материалы.

#### 6.1.Примерные темы письменных учебных работ

**6.1.2 Контрольные работы/ рефераты/ индивидуальные задания обучающемуся.**

#### Индивидуальные задания

1. Разработка программы для построения графиков функций.

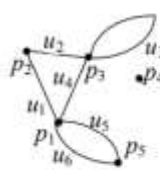


2. Разработка программы для преобразования чисел в разные системы счисления.
3. Разработка программы для шифрования и расшифровки текста.
4. Разработка программы для оценки площади различных фигур.
5. Разработка программы для оценки объема различных фигур.
6. Разработка программы-переводчика и толкового словаря.
7. Разработка программы для подсчета потраченных и потребляемых калорий.
8. Разработка программы-игры в крестики-нолики.
9. Разработка планировщика (списка дел).
10. Разработка программы-будильника и учета времени сна и статистики по нему.
11. Разработка программы для построения трехмерного графика-поверхности.
12. Разработка толкового словаря.
13. Разработка программы для нахождения локального минимума/максимума функции.
14. Разработка программы-будильника.
15. Разработка программы для транслитерации текста.
16. Разработка текстового редактора.
17. Разработка графического редактора.
18. Разработка программы для организации интервальной работы.

## 6.2. Примерные вопросы и задания / задачи для промежуточной аттестации

Форма промежуточной аттестации зачет

Таблица 5 – Типовые (примерные) контрольные вопросы и задания

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания и (или) задачи
<b>Семестр 1 Зачет</b>		
<b>Разделы дисциплины</b>		
1. Дискретная математика с элементами математической логики		
1.1 Введение в теорию множеств.	1. Понятие множества. Операции над множествами. 2. Эквивалентные множества. Мощность множества. Счетное множество и континуум 3. Сравнение мощностей множеств	1. Сколькими способами можно расселить 15 гостей в четырёх комнатах, если требуется, чтобы ни одна из комнат не осталась пустой? 2. Пусть дано некое множество попарно непересекающихся отрезков на числовой прямой. Докажите, что такое множество конечно или счетно. 3. Докажите, что если $M$ – произвольное бесконечное множество и $A$ – счетное множество, то $A \cup M \sim M$ .
1.2 Классическая логика высказываний	1. Принципы построения формализованного языка 2. Общая характеристика и особенности языка	1. Определить, являются ли следующие последовательности символов формулами: а) $\neg(\neg p)$ , б) $\neg(\neg p \vee \neg q)$ , в) $p \& q \vee r$ , г) $(p \& q) \vee r$ , д) $\neg(pq)$ , е) $\neg\neg\neg p$ , ж) $(p \vee \neg q)$ , з) $(A \& B)$ . 2. Установить, являются ли следующие высказывания логически истинными, логически ложными или логически недетерминированными: а) Либо Иван любит Марию, но она его не любит, либо Мария любит Ивана, но он её не любит. б) Число делится на 2 или не делится на 3, если и только если неверно, что когда оно делится на 3, то делится и на 2. в) Если сложное высказывание не относится ни к конъюнктивным, ни к дизъюнктивным, ни к импликативным, то нельзя сказать, что оно конъюнктивное или импликативное. 5. Зададим понятие логического следования
1.3 Введение в теорию графов	1. История возникновения. Основные понятия и их пояснение на примере. 2. Первый способ аналитического задания графа в виде перечня подмножеств вершин 3. Второй способ аналитического задания графа с помощью матрицы инцидентности. Понятия смежности и инцидентности. Принцип изоморфизма. Матрица смежности	1. Построить матрицу инцидентности для неориентированного графа. <div style="text-align: right;">Ответ:</div>  $A = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$
		2. Построить матрицу инцидентности для ориентированного графа.

	<p>4. Элементы графа. Лемма о рукопожатиях. Маршрут графа. Цепь. Цикл. Путь и контур. Связный граф. Полный граф. Турнир. Плоские и планарные графы. Задача о трёх домах и трёх колодцах</p>	 $A = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & -1 \\ -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{pmatrix}$
<p>2. Теория вероятностей и математическая статистика</p>		
<p>2.1 Теория вероятностей</p>	<p>1. Элементарная теория вероятностей. Вероятностное пространство. Условная вероятность и независимость событий. Случайные величины и функции распределения. Построение вероятностных моделей с помощью функций распределения 2. Характеристики распределения случайной величины. Классификация распределений. Предельные теоремы в схеме испытаний Бернулли. Нормальное распределение. Векторные случайные величины. Независимость случайных величин</p>	<p>1. Абонент забыл последнюю цифру номера телефона и поэтому набирает её наугад. Определить вероятность того, что ему придётся звонить не более чем в 3 места. 2. Устройство, состоящее из пяти независимо работающих элементов, включается за время T. Вероятность отказа каждого из них за это время равна 0,2. Найти вероятность того, что откажут: а) три элемента; б) не менее четырех элементов; в) хотя бы один элемент.</p>
<p>2.2 Математическая статистика</p>	<p>1. Основная теорема статистики 2. Исследование выборки</p>	<p>1. Из генеральной совокупности извлечена выборка объема n. Найти выборочную среднюю, выборочную дисперсию, выборочное среднее квадратическое отклонение, исправленную выборочную дисперсию, коэффициент вариации, моду и медиану. 10,5 11 11,5 12 12,5 13 13,5 2 18 40 25 6 5 4 2. Утверждается, что шарики для подшипников, изготовленные автоматическим станком, имеют средний диаметр 10 мм. Используя односторонний критерий с <math>\alpha=0,05</math>, проверить эту гипотезу, если в выборке из n шариков средний диаметр оказался равным 10,3 мм, а дисперсия известна и равна 1 мм.</p>
<p>3. Основы алгоритмизации и программирования</p>		
<p>3.1 Основы алгоритмизации</p>	<p>1. Основные принципы алгоритмизации. Алгоритмы и программы. Данные. Понятие типа данных. Логические основы алгоритмизации.</p>	<p>1. Построить и протестировать блок-схему для задачи: Дано: <math>n=10, x_i=i^2</math>, где <math>i=1, 2, 3, \dots, n</math>. Найти сумму <math>\sum_{i=1}^n x_i</math>.</p>
<p>3.2 Основы программирования</p>	<p>1. Основные принципы программирования. Языки</p>	<p>1.</p>

	<p>программирования: эволюция, классификация. Системы программирования. Файлы данных.</p>	<p><b>Задача: Степень двойки</b></p> <p>Имя входного файла: input.txt  Имя выходного файла: output.txt  Ограничение по времени: 2 секунды  Ограничение по памяти: 64 мегабайта</p> <p>Дано целое неотрицательное число. Необходимо вывести, является ли оно степенью двойки.</p> <p><b>Формат входного файла</b>  В первой строке входного файла подается целое число <math>N</math>, <math>0 \leq N \leq 3024</math>.</p> <p><b>Формат выходного файла</b>  Вывести YES, если число является степенью двойки, NO - иначе.</p> <p><b>Примеры</b></p> <table border="1" data-bbox="1023 387 1505 454"> <thead> <tr> <th>input.txt</th> <th>output.txt</th> </tr> </thead> <tbody> <tr> <td>8</td> <td>YES</td> </tr> <tr> <td>22</td> <td>NO</td> </tr> </tbody> </table>	input.txt	output.txt	8	YES	22	NO
input.txt	output.txt							
8	YES							
22	NO							
<p>3.3 Основы программирования на языке Python</p>	<p>1. История языка C. Взаимоотношения с другими языками. Достоинства языка  2. История языка. Недостатки языка.</p>	<p>1. Обозначьте порядок вычисления выражения по операциям:  <math>11 * 2 ** 2 - 13 / 4 + 7</math>.  Какое целое число получим в итоге?  2. Напишите функцию search_substr(subst, st), которая принимает 2 строки и определяет, имеется ли подстрока subst в строке st. В случае нахождения подстроки, возвращается фраза «Есть контакт!», а иначе «Мимо!».  Должно быть найдено совпадение независимо от регистра обеих строк.</p>						
<p>3.4 Объектно-ориентированное программирование на языке Python</p>	<p>1. Сравнение ООП и процедурного программирования  2. Преимущества ООП</p>	<p>1. Создайте класс Soda (для определения типа газированной воды), принимающий 1 аргумент при инициализации (отвечающий за добавку к выбираемому лимонаду).  В этом классе реализуйте метод show_my_drink(), выводящий на печать «Газировка и {ДОБАВКА}» в случае наличия добавки, а иначе отобразится следующая фраза: «Обычная газировка».  2. Николаю требуется проверить, возможно ли из представленных отрезков условной длины сформировать треугольник.  Для этого он решил создать класс TriangleChecker, принимающий только положительные числа.  С помощью метода is_triangle() возвращаются следующие значения (в зависимости от ситуации):  – Ура, можно построить треугольник!;  – С отрицательными числами ничего не выйдет!;  – Нужно вводить только числа!;  – Жаль, но из этого треугольник не сделать.</p>						
<b>Компетенции</b>								
ОПК - 2		Кейс-задание Разработка игры «Змейка».						
		Кейс-задание Разработка игры «Рас-ман».						

Составитель (и): \_\_\_\_\_

