


МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Ярославский государственный университет им. П.Г. Демидова»



**УТВЕРЖДАЮ**

Проректор по учебной работе

 И.А. Кузнецова

«03» мая 2023 г.

**Дополнительная профессиональная программа  
(программа профессиональной переподготовки)**

**Современное программирование и введение в базы данных**

(наименование программы)

**Информационно-коммуникационные технологии**

(отраслевая принадлежность программы)

2023 г.

## I. Общие положения

1. Дополнительная профессиональная программа (программа профессиональной переподготовки) ИТ-профиля **«Современное программирование и введение в базы данных»** (далее – Программа) разработана в соответствии с нормами Федерального закона РФ от 29 декабря 2012 года № 273-ФЗ «Об образовании в Российской Федерации», с учетом требований приказа Минобрнауки России от 1 июля 2013 г. № 499 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам», с изменениями, внесенными приказом Минобрнауки России от 15 ноября 2013 г. № 1244 «О внесении изменений в Порядок организации и осуществления образовательной деятельности по дополнительным профессиональным программам, утвержденный приказом Министерства образования и науки Российской Федерации от 1 июля 2013 г. № 499», *приказа Министерства образования и науки РФ от 23 августа 2017 г. N 816 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ» (указать при необходимости)*; паспорта федерального проекта «Развитие кадрового потенциала ИТ-отрасли» национальной программы «Цифровая экономика Российской Федерации»; постановления Правительства Российской Федерации от 13 мая 2021 г. № 729 «О мерах по реализации программы стратегического лидерства «Приоритет-2030» (в редакции постановления Правительства Российской Федерации от 14 марта 2022 г. № 357 «О внесении изменений в постановление Правительства Российской Федерации от 13 мая 2021 г. № 729»); приказа Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации от 28 февраля 2022 г. № 143 «Об утверждении методик расчета показателей федеральных проектов национальной программы «Цифровая экономика Российской Федерации» и признании утратившими силу некоторых приказов Министерства цифрового

развития, связи и массовых коммуникаций Российской Федерации об утверждении методик расчета показателей федеральных проектов национальной программы «Цифровая экономика Российской Федерации» (далее – приказ Министерства цифрового развития, связи и массовых коммуникаций Российской Федерации № 143); федеральных государственных образовательных стандартов:

- Высшего образования по направлению подготовки 03.03.02 Физика (уровень бакалавриата), утвержденного приказом Минобрнауки России от 7 августа 2020 г. №891
- Высшего образования по направлению подготовки 03.03.03 Радиофизика (уровень бакалавриата), утвержденного приказом Минобрнауки России от 7 августа 2020 г. №912
- Высшего образования по направлению подготовки 04.03.01 Химия (уровень бакалавриата), утвержденного приказом Минобрнауки России от 17 июля 2017 г. №671
- Высшего образования по направлению подготовки 05.03.06 Экология и природопользование (уровень бакалавриата), утвержденного приказом Минобрнауки России от 7 августа 2020 г. №894
- Высшего образования по направлению подготовки 06.03.01 Биология (уровень бакалавриата), утвержденного приказом Минобрнауки России от 7 августа 2020 г. №920
- Высшего образования по направлению подготовки 37.03.01 Психология (уровень бакалавриата), утвержденного приказом Минобрнауки России от 29 июля 2020 г. №839
- Высшего образования по направлению подготовки 38.03.01 Экономика (уровень бакалавриата), утвержденного приказом Минобрнауки России от 12 августа 2020 г. №954
- Высшего образования по направлению подготовки 38.03.02 Менеджмент (уровень бакалавриата), утвержденного приказом Минобрнауки России от 12 августа 2020 г. №970

- Высшего образования по направлению подготовки 38.03.04 Государственное и муниципальное управление (уровень бакалавриата), утвержденного приказом Минобрнауки России от 13 августа 2020 г. №1016
- Высшего образования по направлению подготовки 39.03.01 Социология (уровень бакалавриата), утвержденного приказом Минобрнауки России от 5 февраля 2018 г. №75
- Высшего образования по направлению подготовки 39.03.02 Социальная работа (уровень бакалавриата), утвержденного приказом Минобрнауки России от 5 февраля 2018 г. №76
- Высшего образования по направлению подготовки 39.03.03 Организация работы с молодежью (уровень бакалавриата), утвержденного приказом Минобрнауки России от 5 февраля 2018 г. №77
- Высшего образования по направлению подготовки 40.03.01 Юриспруденция (уровень бакалавриата), утвержденного приказом Минобрнауки России от 13 августа 2020 г. №1011
- Высшего образования по направлению подготовки 41.03.04 Политология (уровень бакалавриата), утвержденного приказом Минобрнауки России от 23 августа 2017 г. №814
- Высшего образования по направлению подготовки 41.03.06 Публичная политика и социальные науки (уровень бакалавриата), утвержденного приказом Минобрнауки России от 13 августа 2020 г. №1001
- Высшего образования по направлению подготовки 42.03.01 Реклама и связи с общественностью (уровень бакалавриата), утвержденного приказом Минобрнауки России от 8 июня 2017 г. №512
- Высшего образования по направлению подготовки 43.03.02 Туризм (уровень бакалавриата), утвержденного приказом Минобрнауки России от 8 июня 2017 г. №516

- Высшего образования по направлению подготовки 45.03.01 Филология (уровень бакалавриата), утвержденного приказом Минобрнауки России от 12 августа 2020 г. №986
- Высшего образования по направлению подготовки 46.03.01 История (уровень бакалавриата), утвержденного приказом Минобрнауки России от 8 октября 2020 г. №1291
- Высшего образования по направлению подготовки 29.03.01 Технология изделий легкой промышленности (уровень бакалавриата), утвержденного приказом Минобрнауки России от 11 августа 2016 г. №1008
- Высшего образования по направлению подготовки 45.03.02 Лингвистика (уровень бакалавриата), утвержденного приказом Минобрнауки России от 7 августа 2014 г. №940
- Высшего образования по направлению подготовки 07.03.03 Дизайн архитектурной среды (уровень бакалавриата), утвержденного приказом Минобрнауки России от 21 марта 2016 г. №247
- Высшего образования по направлению подготовки 08.03.01 Строительство (уровень бакалавриата), утвержденного приказом Минобрнауки России от 12 марта 2015 г. №201
- Высшего образования по направлению подготовки 22.03.02 Металлургия (уровень бакалавриата), утвержденного приказом Минобрнауки России от 4 декабря 2015 г. №1427
- Высшего образования по направлению подготовки 23.03.03 Эксплуатация транспортно-технологических машин и комплексов (уровень бакалавриата), утвержденного приказом Минобрнауки России от 14 декабря 2015 г. №1470
- Высшего образования по направлению подготовки 38.03.03 Управление персоналом (уровень бакалавриата), утвержденного приказом Минобрнауки России от 14 декабря 2015 г. №1461
- Высшего образования по направлению подготовки 42.03.02 Журналистика (уровень бакалавриата), утвержденного приказом

Минобрнауки России от 7 августа 2014 г. №951

- Высшего образования по направлению подготовки 49.03.01 Физическая культура (уровень бакалавриата), утвержденного приказом Минобрнауки России от 7 августа 2014 г. №935
- Высшего образования по направлению подготовки 51.03.05 Режиссура театрализованных представлений и праздников (уровень бакалавриата), утвержденного приказом Минобрнауки России от 12 марта 2015 г. №205
- Высшего образования по направлению подготовки 03.04.02 Физика (уровень магистратуры), утвержденного приказом Минобрнауки России от 7 августа 2020 г. №914
- Высшего образования по направлению подготовки 03.04.03 Радиофизика (уровень магистратуры), утвержденного приказом Минобрнауки России от 7 августа 2020 г. №918
- Высшего образования по направлению подготовки 04.04.01 Химия (уровень магистратуры), утвержденного приказом Минобрнауки России от 13 июля 2017 г. №655
- Высшего образования по направлению подготовки 05.04.06 Экология и природопользование (уровень магистратуры), утвержденного приказом Минобрнауки России от 7 августа 2020 г. №897
- Высшего образования по направлению подготовки 06.04.01 Биология (уровень магистратуры), утвержденного приказом Минобрнауки России от 11 августа 2020 г. №934
- Высшего образования по направлению подготовки 37.04.01 Психология (уровень магистратуры), утвержденного приказом Минобрнауки России от 29 июля 2020 г. №841
- Высшего образования по направлению подготовки 38.04.01 Экономика (уровень магистратуры), утвержденного приказом Минобрнауки России от 11 августа 2020 г. №939
- Высшего образования по направлению подготовки 38.04.02

Менеджмент (уровень магистратуры), утвержденного приказом Минобрнауки России от 12 августа 2020 г. №952

- Высшего образования по направлению подготовки 38.04.04 Государственное и муниципальное управление (уровень магистратуры), утвержденного приказом Минобрнауки России от 13 августа 2020 г. №1000
- Высшего образования по направлению подготовки 38.04.08 Финансы и кредит (уровень магистратуры), утвержденного приказом Минобрнауки России от 12 августа 2020 г. №991
- Высшего образования по направлению подготовки 39.04.01 Социология (уровень магистратуры), утвержденного приказом Минобрнауки России от 5 февраля 2018 г. №79
- Высшего образования по направлению подготовки 39.04.02 Социальная работа (уровень магистратуры), утвержденного приказом Минобрнауки России от 5 февраля 2018 г. №80
- Высшего образования по направлению подготовки 39.04.03 Организация работы с молодежью (уровень магистратуры), утвержденного приказом Минобрнауки России от 5 февраля 2018 г. №82
- Высшего образования по направлению подготовки 40.04.01 Юриспруденция (уровень магистратуры), утвержденного приказом Минобрнауки России от 25 ноября 2020 г. №1451
- Высшего образования по направлению подготовки 41.04.04 Политология (уровень магистратуры), утвержденного приказом Минобрнауки России от 13 июля 2017 г. №654
- Высшего образования по направлению подготовки 43.04.02 Туризм (уровень магистратуры), утвержденного приказом Минобрнауки России от 15 июня 2017 г. №556
- Высшего образования по направлению подготовки 44.04.02 Психолого-педагогическое образование (уровень магистратуры), утвержденного приказом Минобрнауки России от 22 февраля 2018

г. №127

- Высшего образования по направлению подготовки 45.04.01 Филология (уровень магистратуры), утвержденного приказом Минобрнауки России от 12 августа 2020 г. №980
- Высшего образования по направлению подготовки 46.04.01 История (уровень магистратуры), утвержденного приказом Минобрнауки России от 18 августа 2020 г. №1057
- Высшего образования по направлению подготовки 22.04.02 Metallургия (уровень магистратуры), утвержденного приказом Минобрнауки России от 30 марта 2015 г. №300
- Высшего образования по направлению подготовки 41.04.05 Международные отношения (уровень магистратуры), утвержденного приказом Минобрнауки России от 5 февраля 2018 г. №79
- Высшего образования по направлению подготовки 48.04.01 Теология (уровень магистратуры), утвержденного приказом Минобрнауки России от 17 февраля 2014 г. №125
- Высшего образования по направлению подготовки 54.04.01 Дизайн (уровень магистратуры), утвержденного приказом Минобрнауки России от 21 марта 2016 г. №255
- Высшего образования по направлению подготовки 50.04.01 Искусства и гуманитарные науки (уровень магистратуры), утвержденного приказом Минобрнауки России от 3 декабря 2015 г. №1410

(далее вместе – ФГОС ВО)), а также профессионального стандарта **«Программист»**, утвержденного приказом Министерства труда и социальной защиты РФ от 20 июля 2022 г. № 424н.

2. Профессиональная переподготовка заинтересованных лиц (далее – Слушатели), осуществляемая в соответствии с Программой (далее – Подготовка), имеющей отраслевую направленность **Информационно-коммуникационные технологии**, проводится в ФГБОУ ВО «Ярославский



государственный университет им. П.Г. Демидова» (далее – Университет) в соответствии с учебным планом в очной/очно-заочной форме обучения.

3. Разделы, включенные в учебный план Программы, используются для последующей разработки календарного учебного графика, учебно-тематического плана, рабочей программы, оценочных и методических материалов. Перечисленные документы разрабатываются Университетом самостоятельно, с учетом актуальных положений законодательства об образовании, законодательства в области информационных технологий и смежных областей знаний ФГОС ВО и профессионального стандарта **«Программист»**.

4. Программа регламентирует требования к профессиональной переподготовке в области создание алгоритмов и программ, пригодных для практического применения.

Срок освоения Программы составляет 288 часов.

К освоению Программы в рамках проекта допускаются лица:

- получающие высшее образование по очной (очно-заочной) форме, лица, освоившие основную профессиональную образовательную программу (далее – ОПОП ВО) бакалавриата – в объеме не менее первого курса (бакалавры 2-го курса), ОПОП ВО специалитета – не менее первого и второго курсов (специалисты 3-го курса). Также к освоению ДПП ПП допускаются лица, обучающиеся по программам магистратуры, которые не относятся к ИТ-профилю (согласно приложению к Методике расчета показателя граждан, прошедших обучение по дополнительным образовательным программам) и по программам ординатуры.

5. Область профессиональной деятельности - 06 Связь, информационные и коммуникационные технологии.

## **II. Цель**

6. Целью подготовки слушателей по Программе является получение

компетенции<sup>1</sup>, необходимой для выполнения нового вида профессиональной деятельности в области информационных технологий для обучающихся по специальностям и направлениям подготовки, не отнесенным к ИТ-сфере; приобретение новой квалификации «**Инженер-программист**».

### **III. Характеристика новой квалификации и связанных с ней видов профессиональной деятельности, трудовых функций и (или) уровней квалификации**

7. Виды профессиональной деятельности, трудовая функция, указанные в профессиональном стандарте по соответствующей должности **Инженер-программист**, представлены в таблице 1:

---

<sup>1</sup>Указать целевые группы обучающихся, определенные паспортом Федерального проекта: – обучающиеся по специальностям и направлениям подготовки, не отнесенным к ИТ-сфере, – обучающиеся по специальностям и направлениям подготовки ИТ-сферы (выбрать нужное)

Таблица 1

**Характеристика новой квалификации, связанной с видом профессиональной деятельности и трудовыми функциями в соответствии с профессиональным стандартом «Программист»**

Область профессиональной деятельности	Тип профессиональной деятельности	Код и наименование профессиональной компетенции	Трудовые действия	Трудовая функция	Обобщенная трудовая функция	Вид профессиональной деятельности
Связь, информационные и коммуникационные технологии	Проектный тип профессиональной деятельности	ПК -1. Применяет языки программирования для решения профессиональных задач	Составление формализованных описаний решений поставленных задач в соответствии с требованиями технического задания или внутренних документов организации. Разработка алгоритмов решения поставленных задач в соответствии с требованиями технического задания или внутренних документов организации.	А/01.3 Формализация и алгоритмизация поставленных задач для разработки программного кода	Разработка и отладка программного кода	Разработка компьютерного программного обеспечения
			Создание программного кода в соответствии с техническим заданием (готовыми спецификациями). Оптимизация программного кода с использованием специализированных программных средств.	А/02.3 Написание программного кода с использованием языков программирования, определения и		

			<p>Оценка и согласование сроков выполнения поставленных задач.</p>	<p>манипулирование данными в базах данных</p>		
			<p>Приведение наименований переменных, функций, классов, структур данных и файлов в соответствие с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного кода.</p> <p>Структурирование исходного программного кода в соответствии с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного кода.</p> <p>Комментирование и разметка программного кода в соответствии с нормативно-техническими документами (стандартами и регламентами),</p>	<p>A/03.3 Оформление программного кода в соответствии с установленным и требованиями</p>		

			<p>определяющими требования к оформлению программного кода. Форматирование исходного программного кода в соответствии с нормативно-техническими документами (стандартами и регламентами), определяющими требования к оформлению программного кода.</p>			
			<p>Анализ и проверка исходного программного кода. Отладка программного кода на уровне программных модулей. Отладка программного кода на уровне межмодульных взаимодействий и взаимодействий с окружением. Оценка и согласование сроков выполнения поставленных задач.</p>	<p>A/05.3 Проверка и отладка программного кода</p>		
		<p>ПК-2. Использует СУБД при разработке ПО</p>	<p>Составление формализованных описаний решений поставленных задач в соответствии с требованиями технического задания или внутренних документов организации.</p>	<p>A/01.3 Формализация и алгоритмизация поставленных задач для разработки</p>		

				программного кода		
			Создание программного кода в соответствии с техническим заданием (готовыми спецификациями). Оценка и согласование сроков выполнения поставленных задач.	A/02.3 Написание программного кода с использовани ем языков программирова ния, определения и манипулирован ия данными в базах данных		

Таблица 2

**Характеристика новой и развиваемой цифровой компетенции в ИТ-сфере, связанной с уровнем формирования и развития в результате освоения Программы «Современное программирование и введение в базы данных»**

Наименование сферы	Код и наименование профессиональной компетенции	Пример инструментов	0 — способность не проявляется/ проявляется в степени, недостаточной для отнесения к 1 уровню сформированности компетенции	1 — способность проявляется под внешним контролем / при внешней постановке задачи/ обучающийся пользуется готовыми, рекомендованным и продуктами	2 — способность проявляется, но обучающийся эпизодически прибегает к экспертной консультации/ самостоятельно подбирает и пользуется готовыми продуктами	3 — способность проявляется системно / обучающийся модифицирует способность под определенные задачи / создает новый продукт, обучает других
Средства программной разработки	ПК -1. Применяет языки программирования для решения профессиональных задач	Python	(-)	(+)	(-)	(-)

<b>Средства программ ной разработк и</b>	ПК-2. Использует СУБД при разработке ПО	MySQL	(-)	(+)	(-)	(-)
----------------------------------------------------------	-----------------------------------------------	-------	-----	-----	-----	-----



#### **IV. Характеристика новых и развиваемых цифровых компетенций, формирующихся в результате освоения программы**

8. В ходе освоения Программы Слушателем приобретаются следующие профессиональные компетенции:

- ПК-1 Применяет языки программирования для решения профессиональных задач (id компетенции 28 согласно Требованиям к дополнительным профессиональным программам в рамках проекта «Цифровые кафедры»);

- ПК-2. Использует СУБД при разработке ПО (id компетенции 32 согласно Требованиям к дополнительным профессиональным программам в рамках проекта «Цифровые кафедры»);

#### **V. Планируемые результаты обучения по ДПП III**

10. Результатами подготовки слушателей по Программе является получение компетенции, необходимой для выполнения нового вида профессиональной деятельности в области информационных технологий — создание алгоритмов и программ, пригодных для практического применения; приобретение новой квалификации «Инженер-программист».

**Наименование компетенции: ПК-1 Применяет языки программирования для решения профессиональных задач**

*Знать:*

Методы и приемы формализации поставленных задач;

Методы и приемы алгоритмизации задач

Алгоритмы решения типовых задач, области и способы их применения

*Уметь:*

Использовать методы и приемы формализации задач

Использовать методы и приемы алгоритмизации поставленных задач

Использовать выбранную среду программирования и средства системы управления базами данных

Использовать возможности имеющейся технической и/или программной архитектуры

*Иметь навыки:*

Применять современные компиляторы, отладчики и оптимизаторы программного кода

Применять стандартные алгоритмы в соответствующих областях

Применять выбранные языки программирования для написания программного кода

**Наименование компетенции: ПК-2. Использует СУБД при разработке ПО**

*Знать:*

Технологии программирования;

Методологии и технологии проектирования и использования баз данных

*Уметь:*

Использовать выбранную среду программирования и средства системы управления базами данных;

*Иметь навыки:*

Особенности выбранной среды программирования и системы управления базами данных

## **VI. Организационно-педагогические условия реализации ДПП**

12. Реализация Программы должна обеспечить получение компетенции, необходимой для выполнения нового вида профессиональной деятельности в области информационных технологий — создание алгоритмов и программ, пригодных для практического применения; приобретение новой квалификации «Инженер-программист».

13. Учебный процесс организуется с применением электронного обучения и дистанционных образовательных технологий, инновационных технологий и методик обучения, способных обеспечить получение слушателями знаний, умений и навыков в области информационно-коммуникационных технологий (другие области профессиональной деятельности).

14. Реализация Программы обеспечивается научно-педагогическими кадрами Университета, допустимо привлечение к образовательному процессу высококвалифицированных специалистов ИТ-сферы и/или дополнительного профессионального образования в части, касающейся профессиональных компетенций в области создания алгоритмов и программ, пригодных для практического применения, с обязательным участием представителей профильных организаций-работодателей. Возможно привлечение региональных руководителей цифровой трансформации (отраслевых ведомственных и/или корпоративных) к проведению итоговой аттестации, привлечение работников организаций реального сектора экономики субъектов Российской Федерации.

## VII. Учебный план ДПП

15. Объем Программы составляет 288 часов.

16. Учебный план Программы определяет перечень, последовательность, общую трудоемкость разделов и формы контроля знаний.

Учебный план программы профессиональной переподготовки  
«Современное программирование и введение в базы данных»

№ п/п	Наименование раздела (модуля)	Общая трудоемкость (288 часов)	Форма контроля
1.	Введение в информатику и программирование	24	Итоговый тест по разделу
2.	Основы программирования на языке Python	51	Итоговый экзамен по разделу

3.	Реализация алгоритмов на языке Python	36	Итоговый экзамен по разделу
4.	Введение в базы данных	72	Итоговый экзамен по разделу
5.	Практика	60	Отчет о практике
	Промежуточная аттестация	9	Итоговые контрольные мероприятия по разделам
	Итоговая аттестация	36	Защита итогового проекта
	Итого:	288	





4	Введение в базы данных	10										
5	Практика			6	6	6	6	6	6	6	6	6
6	Промежуточная аттестация		8									
7	Итоговая аттестация											

№ п/п	Наименование раздела (модуля)	Учебная неделя				
		32	33	34	35	36
1	Введение в информатику и программирование					
2	Основы программирования на языке Python					
3	Реализация алгоритмов на языке Python					
4	Введение в базы данных					
5	Практика	6				
6	Промежуточная аттестация	2				
7	Итоговая аттестация		9	9	9	9

## IX. Рабочая программа учебных предметов, курсов, дисциплин (модулей)

19. Рабочая программа содержит перечень разделов и тем, а также рассматриваемых в них вопросов с учетом их трудоемкости.

Рабочая программа разрабатывается Университетом с учетом профессионального стандарта «Программист».

№ п/п	Наименование и краткое содержание раздела(модуля)	Объем, часов
1.	<p>Раздел 1. Введение в информатику и программирование</p> <p><i>Основные темы:</i></p> <p>Тема 1.1. Теоретические основы информатики</p> <p>Тема 1.2. Техническое и программное обеспечение</p> <p>Тема 1.3. Хранение данных</p> <p>Тема 1.4. Общие сведения о языках программирования</p> <p><i>Краткое содержание</i></p> <p>Раздел дает целостное представление об информатике, ее основных понятиях, законах и тенденциях развития современных компьютерных систем. Освоение курса позволит преодолеть возможные препятствия и предубеждения, мешающие уверенному овладению компьютерными технологиями каждым, кто хочет легко ориентироваться в мире информационных технологий и подготовиться к освоению программирования.</p>	24
2.	<p>Раздел 2. Основы программирования на языке Python</p> <p><i>Основные темы:</i></p> <p>Тема 2.1. Константы, переменные, условный оператор</p> <p>Тема 2.2. Глобальные переменные и циклы</p> <p>Тема 2.3. Строки, списки и цикл for</p> <p>Тема 2.4. Словари, кортежи и файлы</p> <p>Тема 2.5. Классы, объекты и ссылки</p> <p><i>Краткое содержание</i></p> <p>Раздел посвящен изучению одного из самых востребованных языков программирования Python, находящего широкий круг приложений. Слушатели детально изучат конструкции языка Python, начиная с простых конструкций условного оператора, цикла и других стандартных конструкций языков программирования, а также разберут основы объектно-ориентированного программирования. Вместе с конструкциями языка программирования разбираются простейшие алгоритмические приемы, необходимые современному программисту.</p> <p>В рамках раздела осуществляется автоматическая проверка задач с использованием системы Яндекс.Контест.</p>	51
3.	Раздел 3. Реализация алгоритмов на языке Python	36



	<p><i>Основные темы:</i>  Тема 3.1. Переборные алгоритмы  Тема 3.2. Графовые алгоритмы  Тема 3.3. Алгоритмические задачи  <i>Краткое содержание</i>  Раздел посвящен развитию навыков алгоритмизации слушателей программы. В рамках раздела подробно разбираются идеи, позволяющие реализовывать переборные алгоритмы, которые являются естественным подходом, использующим преимущество в скорости расчета вычислительной техники перед человеком. Рассматриваются графовые алгоритмы, которые являются простой, но в то же время очень мощной моделью, широко используемой на практике. И, наконец, дается введение в алгоритмы класса «разделяй-и-властвуй» и динамического программирования, которые позволяют учесть структуру задач для построения более эффективного решения. Это позволяет применять современные идеи для решения широкого класса задач.  В рамках раздела осуществляется автоматическая проверка задач с использованием системы Яндекс.Контест.</p>	
4.	<p>Раздел 4. Введение в базы данных  <i>Основные темы:</i>  Тема 4.1. Введение в базы данных. Таблицы  Тема 4.2. Запросы на выборку данных  Тема 4.3. Запросы на изменение данных  Тема 4.4. Сложные запросы с использованием нескольких таблиц  Тема 4.5. Проектирование баз данных  Тема 4.6. Введение в ORM  <i>Краткое содержание</i>  В разделе дается системное изложение теории реляционных баз данных и практическое применение реляционных баз данных в СУБД PostgreSQL. Раздел включает в себя темы, связанные с принципами организации баз данных в виде системы взаимосвязанных таблиц, подходами к построению запросов на выборку, изменение, добавление и удаление данных. Рассматриваются вопросы построения сложных запросов с использованием нескольких таблиц. Для разработки собственных решений на основе баз данных важными являются темы по проектированию баз данных и технологии Object-Relational Mapping (ORM).  В рамках раздела применяется автоматическая проверка задач с использованием системы Яндекс.Контест.</p>	72
5.	<p><i>Основные темы: Практика</i>  <i>Краткое содержание:</i>  Слушатели программы проходят практику на базе профильных ИТ-компаний, что позволяет развить овладение информационными технологиями в области создания алгоритмов и программ, пригодных для практического применения. Практика предполагает решение задач, имеющих практическую направленность и учитывающих современное развитие информационных технологий. По результатам практики обучающиеся должны сделать прототип программного проекта.</p>	60
и	<p><i>Основные темы: Итоговая аттестация</i></p>	36

т.д.	<i>Краткое содержание:</i> В рамках итоговой аттестации слушатели выполняют проект, имеющий практическую направленность. Результаты выполнения проекта представляются к защите и позволяют оценить степень развития профессиональных компетенций.	
------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

20. Учебно-тематический план Программы определяет тематическое содержание, последовательность разделов и (или) тем и их трудоемкость.

№ п/п	Наименование раздела (модуля)	Количество часов		
		аудиторных		самостоятельной работы (выполнение заданий для самостоятельной работы, тестовых заданий, констестов, заданий для промежуточной аттестации)
		Лекции	Семинары	
1.	Введение в информатику и программирование	8	8	8
2.	Основы программирования на языке Python	10	10	31
3.	Реализация алгоритмов на языке Python	6	6	24
4.	Введение в базы данных	12	12	48
5.	Практика			60
6.	<b>Промежуточная аттестация</b>	9		
	<b>Итоговая аттестация</b>	36		

*\*указать вид (-ы) запланированной самостоятельной работы*

## X. Формы аттестации

21. Слушатели, успешно выполнившие все элементы учебного плана, допускаются к итоговой аттестации.

Итоговая аттестация по Программе проводится в форме

демонстрационного экзамена.

22. Лицам, успешно освоившим Программу (в области создания алгоритмов и программ, пригодных для практического применения, или навыков использования и освоения цифровых технологий, необходимых для выполнения нового вида профессиональной деятельности) и прошедшим итоговую аттестацию в рамках проекта «Цифровые кафедры», выдается документ о квалификации: диплом о профессиональной переподготовке.

При освоении ДПП ПП параллельно с получением высшего образования диплом о профессиональной переподготовке выдается не ранее получения соответствующего документа об образовании и о квалификации (за исключением лиц, имеющих среднее профессиональное или высшее образование).

23. Лицам, не прошедшим итоговую аттестацию или получившим на итоговой аттестации неудовлетворительные результаты, а также лицам, освоившим часть Программы и (или) отчисленным из Университета, выдается справка об обучении или о периоде обучения по образцу, самостоятельно устанавливаемому Университетом.

## **XI. Оценочные материалы**

24. Контроль знаний, полученных слушателями при освоении разделов (модулей) Программы, осуществляется в следующих формах:

- текущий контроль успеваемости – обеспечивает оценивание хода освоения разделов Программы, проводится в форме тестов и/или контестов (с проверкой в автоматическом режиме), а также иных формах;

- промежуточная аттестация – завершает изучение отдельного раздела Программы, проводится в форме теста и/или контеста (с проверкой в автоматическом режиме), промежуточная аттестация включает успешное прохождение ассесментов;

- итоговая аттестация – завершает изучение всей программы.

25. В ходе освоения Программы каждый слушатель выполняет следующие отчетные работы:

№ п/п	Наименование раздела (модуля)	Задание	Критерии оценки
1.	Введение в информатику и программирование	Тест (п.26.1) Тест (п.26.2) Тест (п.26.3) Тест (п.26.4)	<p>В рамках раздела формируется компетенция ПК-1.</p> <p>В рамках текущего контроля успеваемости оценка «удовлетворительно» ставится при правильном выполнении не менее 40% тестовых заданий в каждом тесте.</p> <p>Оценка «хорошо» ставится при правильном выполнении не менее 65% тестовых заданий.</p> <p>Оценка «отлично» ставится при правильном выполнении не менее 85% тестовых заданий.</p>
2.	Основы программирования на языке Python	Тест (п.26.5) Контест (п.26.6) Тест (п.26.7) Котест (п.26.8) Тест (п.26.9) Контест (п.26.10) Тест (п.26.11) Контест (п.26.12) Тест (п.26.13)	<p>В рамках раздела формируется компетенция ПК-1.</p> <p>В рамках текущего контроля успеваемости оценка «удовлетворительно» ставится при правильном выполнении не менее 40% тестовых заданий в каждом тесте и решении минимум одной задачи в каждом контесте. В случае наличия нерешенных задач в одном или нескольких контестах, это количество должно быть компенсировано решением дополнительных задач в других контестах.</p> <p>Оценка «хорошо» ставится при правильном выполнении не менее 65% тестовых заданий и решении минимум одной задачи в каждом из контестов.</p>

			Оценка «отлично» ставится при правильном выполнении не менее 85% тестовых заданий и решении более половины задач в каждом из контестов.
3.	Реализация алгоритмов на языке Python	Тест (п.26.14) Контест (п.26.15) Тест (п.26.16) Контест (п.26.17) Тест (п.26.18)	<p>В рамках раздела формируется компетенция ПК-1. В рамках текущего контроля успеваемости оценка «удовлетворительно» ставится при правильном выполнении не менее 40% тестовых заданий в каждом тесте и решении минимум одной задачи в каждом контесте. В случае наличия нерешенных задач в одном или нескольких контестах, это количество должно быть компенсировано решением дополнительных задач в других контестах.</p> <p>Оценка «хорошо» ставится при правильном выполнении не менее 65% тестовых заданий и решении минимум одной задачи в каждом из контестов.</p> <p>Оценка «отлично» ставится при правильном выполнении не менее 85% тестовых заданий и решении более половины задач в каждом из контестов.</p>
4.	Введение в базы данных	Тест (п. 26.19) Контест (п. 26.20) Тест и контест (п. 26.21) Тест и контест (п. 26.22)	<p>В рамках раздела формируется компетенция ПК-2. В рамках текущего контроля успеваемости оценка «удовлетворительно» ставится при правильном выполнении не менее 40% тестовых заданий в каждом тесте и решении минимум одной задачи в каждом контесте. В случае наличия нерешенных</p>

		<p>Тест (п. 26.23) Тест (п. 26.24)</p>	<p>задач в одном или нескольких контестах, это количество должно быть компенсировано решением дополнительных задач в других контестах.</p> <p>Оценка «хорошо» ставится при правильном выполнении не менее 65% тестовых заданий и решении минимум одной задачи в каждом из констестов.</p> <p>Оценка «отлично» ставится при правильном выполнении не менее 85% тестовых заданий и решении более половины задач в каждом из констестов.</p>
5.	Практика	Отчет по практике (п. 29)	<p>В рамках практики завершается формирование компетенций ПК-1 и ПК-2.</p> <p>Практика проходит на базе ИТ-компаний, где студентам выдается задание в виде программного проекта и им необходимо за время практики наметить пути реализации проекта и реализовать его прототип.</p> <p>Пороговый уровень развития компетенции ПК-1 достигается, если обучающийся правильно выбрал алгоритмические подходы к решению задачи и сформировал задел в виде программного кода для решения поставленной задачи.</p> <p>Пороговый уровень развития компетенции ПК-2 достигается, если студент в целом сделал проект схемы базы данных для своего проекта.</p> <p>При достижении порогового уровня ставится</p>

			<p>оценка «удовлетворительно».</p> <p>Оценка «хорошо» ставится если в проекте реализовано не менее 50% требований даже при условии отсутствия всестороннего тестирования проекта.</p> <p>Оценка «отлично» ставится если в проекте реализовано не менее 70% требований даже при условии отсутствия всестороннего тестирования проекта.</p>
--	--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## 26. Текущий контроль. Перечень примерных заданий.

### Раздел «Введение в информатику и программирование»

#### 26.1. Тема «Теоретические основы информатики и история компьютерной техники»

1. Данные — это:

- А) представление информации в материальной форме;
- В) знания относительно фактов, событий, вещей, идей и понятий, которые в определённом контексте имеют конкретный смысл;
- С) сведения, воспринимаемые человеком как отражение фактов материального или духовного мира в процессе коммуникации.

2. В шестнадцатичной системе арабские цифры дополнены следующими символами:

- А) А, В, С;
- В) А, В, С, D, E, F;
- С) А, В, С, D, E.

3. Бит может принимать:

- А) два значения: 0 и 1
- В) значения в диапазоне от 0 до 255
- С) значения в диапазоне от -127 до 128

4. В кодировке UTF-8 для представления символа:

- А) используется код переменной длины;
- В) используется фиксированное количество бит;
- С) не используется битовое представление.

5. Известно, что одно sms-сообщение, если его писать латиницей, сможет вместить до 160 знаков, при этом русских (кириллических) букв в него войдёт не более 70. Это связано с

- А) международными санкциями;
- В) особенностями кодирования;
- С) богатством и смысловым разнообразием русского языка.

#### 26.2. Тема «Техническое и программное обеспечение»

1. Материнская плата в компьютере служит:

- А) основой компьютера, к которой присоединяются все остальные устройства;
- В) для порождения процессов в операционной системе;
- С) для обеспечения безопасности, в том числе от поражения пользователя электрическим током.

2. Процессор компьютера является:



- A) вспомогательным (неосновным) устройством;
- B) обычно требует охлаждения;
- C) служит для хранения программ и данных.

3. Нецифровым (аналоговым) способом подключения монитора к компьютеру является:

- A) HDMI;
- B) Display Port;
- C) VGA.

4. Виртуальная память нужна для:

- A) увеличения объема физической памяти;
- B) разграничения адресных пространств процессов;
- C) хранения и обработки виртуальных данных.

5. Объем виртуальной памяти может быть:

- A) обязательно больше объема физической памяти;
- B) обязательно меньше объема физической памяти;
- C) как больше, так и меньше объема физической памяти.

### 26.3. Тема «Хранение данных»

1. В какой операционной системе устройства хранения поименованы латинскими буквами и эти имена используются при указании пути к файлу:

- A) Windows;
- B) Linux;
- C) Mac OS X.

2. Относительный путь к файлу отсчитывается от:

- A) корневого директория (папки);
- B) текущего директория (папки);
- C) директория пользователя (папки).

3. Величина энтропии позволяет судить о:

- A) возможности сжатия данных с потерями;
- B) фрагментации данных на диске;
- C) теоретической границе, до которой можно сжать данные без потерь.

4. Протокол доступа к внешнему хранилищу NAS:

- A) SSD;
- B) SMB;
- C) DNS.

5. Укажите децентрализованную систему синхронизации файлов:

- A) Яндекс Диск;
- B) Syncthing;

C) DropBox.

#### 26.4. Тема «Обеспечение безопасности данных»

1. Энигма — это переносная шифровальная машина:
  - A) использовавшаяся во время гражданской войны в США;
  - B) разработанная советскими инженерами во время Великой Отечественной войны;
  - C) получившая распространение в нацистской Германии во время Второй мировой войны.
2. Криптоанализ — наука о:
  - A) шифровании и алгоритмах шифрования;
  - B) «взломе» шифров;
  - C) движении криптоактивов (криптовалют).
3. При анализе криптографических протоколов, стороны обычно именуется:
  - A) латинскими буквами A, B, C;
  - B) греческими буквами  $\alpha$ ,  $\beta$ ,  $\gamma$ ;
  - C) именами Alice, Bob, Chuck.
4. Минимальное изменение данных (1 бит) приводит к:
  - A) минимальному изменению хэш-значения (1 бит);
  - B) изменению длины (размера) хэш-значения;
  - C) непредсказуемому изменению хэш-значения при сохранении его размера.
5. Инфраструктура открытых ключей PKI помогает:
  - A) выпускать сертификаты открытого ключа;
  - B) противостоять атакам «человек посередине»;
  - C) вести учет симметричных ключей.

#### **Раздел «Основы программирования на языке Python»**

#### 26.5. Тема «Константы, переменные, условный оператор»

1. Какая функция используется в Python для вывода значения на экран?
2. Какое максимальное количество else может соответствовать одной инструкции if ?
  - A) 0
  - B) 1
  - C) 0 и более
3. Какое обычно расширение у файлов с исходным кодом на языке Python?
4. Приведенный ниже код содержит синтаксические ошибки. Исправьте их. После исправления ошибок программа должна напечатать три числа. Первое

из них 1.0, а второе равно -1.61803.

```
def 3cubic_roots(p q):
    c_1 = 2*math.square_root(-p/3)
    d_1 = ((3*q)/(2*p))*sqrt(-3/p)
    k_1 == 2*math.pi/3
    k_2 = 2*pi*2/3
    t_0 = c_1*math.cos(math.acos(d_1)/3)
    t_1 = c_1*math.cos(acos(d_1)/3+k_1)
    t_2 == c_1*cos(math.acos(d_1)/3+k_2)
    print round(t_0_, 5), round(t_1, 5) eound(t_2, 5)

3cubic_roots(-2, 1)
```

5. Функции `random.randint(0, 24)` и `random.randrange(0, 24)` используются для генерации случайных чисел в выбранном диапазоне. Какое значение может быть сгенерировано одной из этих функций, но не может быть сгенерировано другой?

26.6. Тема «Константы, переменные, условный оператор» - контекст

1. Яблоки.

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод
Вывод	стандартный вывод

Компания мальчиков нарвала в саду яблок. Они хотят разделить яблоки между собой поровну, а то что останется, отдать маме на пирог. Определите, сколько яблок достанется каждому из них, а сколько они отдадут маме.

Формат ввода

В первой строке подается число  $N > 0$  - количество мальчиков в компании. В второй строке подается число  $M \geq N$  - количество яблок.

Формат вывода

Вывести через пробел два целых числа: количество яблок, которое достанется каждому из мальчиков, и количество яблок, которые они отдадут маме.

Пример

Ввод	Вывод
3	4 0
12	

2. Треугольники.

Ограничение времени	1 секунда
Ограничение памяти	64Mb

Ввод	стандартный ввод
Вывод	стандартный вывод

На вход подаются три вещественных числа. Напишите программу, которая определяет, возможно ли составить треугольник с длинами сторон, равными этим числам. Если треугольник возможно составить, то определить, является ли он прямоугольным, тупоугольным или остроугольным.

#### Формат ввода

В первой строке подается первое вещественное число. Во второй строке вводится второе вещественное число. В третьей строке вводится третье вещественное число.

#### Формат вывода

Если треугольник нельзя составить, вывести Нельзя. Если составляется прямоугольным треугольником, то вывести строку Прямоугольный, если составляется тупоугольный треугольник, то вывести строку Тупоугольный, иначе вывести строку Остроугольный.

Выводимая строка начинается с заглавной буквы, остальные буквы строчные.

#### Пример

Ввод	Вывод
3.0	прямоугольный
5.0	
4.0	

## 26.7. Тема «Глобальные переменные и цикл while»

### 1. Что делает команда continue?

- A) Выходит из выполняемого цикла;
- B) Останавливает работу программы;
- C) Осуществляет переход к следующей итерации цикла;
- D) Сбрасывает переменную, по которой выполняется итерация, до начального значения.

### 2. Рассмотрим следующий фрагмент кода:

```
a = 1

def fun(x):
    global a
    a += a + 1
    return x * 3 * 4 * 5

a = 134
print(fun(fun(fun(fun(1)))))
```

Каким будет значение переменной a после его выполнения?

3. Пусть у нас есть число  $n$ . Будем производить следующие действия:

А) Если число четное, то поделим его на 2.

В) Если оно нечетное, то умножаем его на 3 и прибавляем 1.

Над полученным числом будем выполнять те же самые действия до тех пор, пока не получим 1.

Для справки, чтобы дойти до единицы от  $n = 11$  требуется 14 шагов, а от  $n = 80$  необходимо сделать 9 шагов.

В 1937 году математик Лотар Коллатц выдвинул гипотезу, которая заключается в том, что какое бы начальное число мы ни взяли, рано или поздно мы получим единицу.

Более подробно про эту гипотезу можно почитать здесь — [https://ru.wikipedia.org/wiki/Гипотеза\\_Коллатца](https://ru.wikipedia.org/wiki/Гипотеза_Коллатца)

Напишите программу, которая вычисляет сколько шагов надо сделать, чтобы дойти до единицы, если исходное  $n = 54$

4. Что делает команда `break`?

А) Осуществляет переход к следующей итерации цикла;

В) Выходит из выполняемого цикла;

С) Прерывает выполнение программы;

Д) Сбрасывает переменную, по которой выполняется итерация, до начального значения

5. Рассмотрим следующий фрагмент кода:

```
r = 42
a = 32

def f(a):
    d = 123*(a+r)
    return d
```

Какие из имен в следующем фрагменте кода являются локальными?

А) `a`;

В) `d`;

С) `f`;

Д) `r`

26.8. Тема «Глобальные переменные и цикл `while`» - контекст

1. Последовательность чисел

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод

Вывод	стандартный вывод
-------	-------------------

На вход программе подается  $N$  элементов ( $1 \leq N < 1000$ ).

Элементы могут принимать целые значения от  $-10\,000$  до  $10\,000$  включительно.

Напишите программу, позволяющую найти и вывести количество пар элементов, в которых хотя бы одно число делится на 7.

В данной задаче под парой подразумевается два подряд идущих элемента.

Формат ввода

Сначала подается число  $N$ , а после него  $N$  чисел, по одному в каждой строке.

Формат вывода

Единственное число, определяющее количество искомых пар.

Пример

Ввод	Вывод
3	0
1	
43	
13	

## 2. Сумма цифр, кратных 4.

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод
Вывод	стандартный вывод

На обработку поступает натуральное число, не превышающее  $10^9$ .

Нужно написать программу, которая выводит на экран сумму цифр числа, которые делятся на четыре. Если в числе нет таких цифр, требуется на экран вывести No.

Формат ввода

Натуральное число  $0 \leq N < 10^9$ .

Формат вывода

Сумма цифр числа, кратных 4. Если таких цифр нет, то No.

Пример

Ввод	Вывод
32	No

## 26.9. Тема «Строки, списки и цикл for»

1. Как использовать оператор для создания среза [x:y] для получения подстроки 'uniyar' из строки

x = 'From [chaly@uniyar.ac.ru](mailto:chaly@uniyar.ac.ru)'?

В качестве ответа введите оператор среза в формате [x:y]

2. Сколько пробелов будет в строке s после выполнения следующего кода:

```
s = """0 сколько нам\тоткрытий \тчудных
готовит просвещения дух"""
result = list()
for w in s.split():
    result.append(w.lower())
s = ' '.join(result)
```

3. Следующий фрагмент кода содержит ошибку:

```
lst = list(range(20))
for a in lst:
    a = str(a)
print('+'.join(lst), end='=')
print(sum(lst))
```

Какое утверждение наилучшим образом описывает эту ошибку?

- A) В этом фрагменте когда неправильно установлены отступы;
- B) В строке 4 попытка объединить в строчку целочисленные значения, а не строковые;
- C) Функция print не может иметь параметр end;
- D) В строке 5 совершена попытка просуммировать строковые значения.

4. Введите строчку, которая будет в переменной mysub после выполнения следующего кода:

```
data = 'accctacgtac'
pos = data.rfind('ta')
mysub = data[pos:pos+4]
```

5. Какая из следующих строковых функций убирает пробельные символы в начале и в конце строки?

- A) rstrip()
- B) split()
- C) strip()
- D) lstrip()
- E) strip()

## 26.10. Тема «Строки, списки и цикл for» - контекст

1. Лампочка на складе

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

На складе сосисок установлена система автоматического включения и выключения освещения, которая настроена таким образом, что она загорается когда на склад заходит человек и выключается когда люди отсутствуют.

Также на складе живет кот Василий, на которого система не реагирует. Когда на складе светло, он делает вид что ловит мышей, однако как только свет выключается он начинает есть сосиски со скоростью одна сосиска в минуту.

В начале рабочего дня известно сколько сосисок находится на складе. Вам надо вывести одно число - сколько сосисок останется на складе к концу рабочего дня, который длится 480 минут: с 0-й по 479-ю минуту включительно.

Формат ввода

В первой строке вводятся через пробел два числа  $0 < K \leq 100000$  и  $N \geq 0$ , которые задают количество интервалов времени, когда кто-то присутствует, и количество сосисок на складе в начале рабочего дня.

Далее следуют  $K$  строк, в каждой из которых через пробел записаны два целых числа  $0 \leq t_1 \leq t_2 < 480$ , которые задают когда человек заходит на склад ( $t_1$ ) и когда он выходит со склада ( $t_2$ ). Эти интервалы следуют в произвольном порядке.

Формат вывода

Единственное число, соответствующее количеству оставшихся сосисок на складе.

Пример

Ввод  
3 40  
0 5  
13 18  
25 475

Вывод  
23

## 2. Самый длинный общий префикс.

Ограничение времени	0.1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод
Вывод	стандартный вывод



Дан набор из N строк. Вам надо найти самый длинный общий префикс среди этих строк.

Формат ввода

На вход подается в первой строке число N, за которым в отдельных строках следуют N строк.

Значение N не превосходит 1000. Длина каждой строки не превосходит 200.

Формат вывода

Вывести самый длинный общий префикс у заданного набора строк.

Пример

Ввод

3

shrimp

shrink

shrine

Вывод

shri

## 26.11. Тема «Словари, кортежи и файлы»

1. Чему должно равняться значение переменной s, чтобы следующая программа вывела на экран 'gagcta'?

```
crypt = {"a": "c", "t": "g", "g": "a", "c": "t"}
s = #####
r = ""
for i in range(len(s)):
    for k, v in crypt.items():
        if v == s[i]:
            r += k
print(r)
```

В качестве ответа приведите значение строки s.

2. Что напечатает следующая программа:

```
stuff = dict()
stuff['orange'] = 'orange'
stuff['apple'] = 'green'
print(stuff['banana'])
```

- A) 0;
- B) None;
- C) yellow;
- D) Программа завершится ошибкой;
- E) -1.

3. Какие методы работают и для списка (list) и для кортежа (tuple)?

Выберите один или несколько ответов:

- A) .append();
- B) .sort();

- C) .index();
- D) .pop();
- E) .count()

4. Чему может равняться значение переменной s, чтобы следующая программа что-нибудь вывела на экран?

```
x = (4, 2, 9)
if s > x:
    print("Bigger!")
```

- A) (4, 2, 9);
- B) (0, 100, 100);
- C) (4, 2);
- D) (4, 2, 9, 0);
- E) (8, 3);
- F) (4, 1, 2);
- G) (10, 1, 1)

5. На этой неделе вы познакомились со словарями (dict), кортежами (tuple) и множествами set. Какие из следующих утверждений являются верными?

- A) Списки сохраняют порядок следования элементов, а кортежи нет;
- B) Длину кортежа можно увеличить после его создания;
- C) Списки проиндексированы целыми числами, а индексом в словаре может быть строка;
- D) Список может быть значением в словаре;
- E) Ключом в словаре может быть список;
- F) Содержимое списка можно изменить, а содержимое кортежа нельзя;
- G) Элементом множества может быть кортеж, строка или целое число, а словарь нет

26.12. Тема «Словари, кортежи и файлы» - контекст

### 1. Кофеманы

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	input.txt
Вывод	output.txt

В кофейни города N приходят посетители и заказывают кофе. Каждый посетитель может приходить несколько раз и делать несколько заказов. Найдите посетителей, которые ходят во все кофейни.

Формат ввода

В файле записаны строки, в каждой строке через пробел указывается название кофейни и имя посетителя.

### Формат вывода

В качестве результата ваша программа должна вывести в выходной файл в список посетителей, которые ходят во все кофейни.

Элементы списка записываются в одну строку через пробел в алфавитном порядке.

В том случае, если таких посетителей нет, необходимо вывести строку No.

### Пример

Ввод

CH Andrew

CH Mike

SB Andrew

CM Mike

CM Andrew

Вывод

Andrew

## 2. Гости

Ограничение времени	0.5 секунд
Ограничение памяти	64Mb
Ввод	input.txt
Вывод	output.txt

В доме несколько подъездов, которые нумеруются начиная с единицы. В каждом подъезде одинаковое, большее единицы количество квартир. Квартиры нумеруются подряд, начиная с единицы.

В доме оборудована система "умный дом" которая регистрирует, в какую квартиру позвонил гость.

В качестве результата ваша программа должна вывести номер подъезда, куда пришло наибольшее количество посетителей. Если таких подъездов несколько, необходимо вывести их все, начиная от наименьшего к наибольшему.

Первые два посетителя приходят в первую и последнюю квартиру одного из подъездов дома.

Иногда, конечно происходят сбои, которые приводят к тому, что второй посетитель неправильно регистрируется, и тогда по журналу регистраций можно увидеть нарушение принципа, что в каждом подъезде одинаковое, большее единицы количество квартир.

Если такое случилось, то надо вывести 'Error' и завершить работу программы.

Если же этот принцип не нарушен, то тогда считается, что первые посетители пришли в точности в первую и последнюю квартиру некоторого подъезда.

### Формат ввода

На вход подается последовательность номеров квартир  $a_i, 1 \leq a_i \leq 10000$ , куда приходят посетители. Количество посетителей не превосходит 1000.

Посетители могут приходить в одну и ту же квартиру несколько раз.

### Формат вывода

В качестве результата ваша программа должна вывести номер подъезда, куда пришло наибольшее количество посетителей.

Если таких подъездов несколько, необходимо вывести их все через пробел, начиная от наименьшего к наибольшему.

### Пример

Ввод	Вывод
1 1	Error

## 26.13. Тема «Классы, объекты и ссылки»

### 1. Рассмотрим следующую программу:

```
a = [49, 27, 101, -10]
b = a
c = list(a)
d = c
a[3] = 68
c[2] = a[1]
b = a[1:3]
b[1] = c[2]
```

Каким будет список в переменной `b` после выполнения этой программы?  
В качестве ответа приведите значение этого списка в виде `[1, 2, 3,...]`

2. Каждый класс должен иметь конструктор, который инициализирует создаваемый объект. Как в Python называется этот конструктор?

Выберите один ответ:

- A) Имя конструктора совпадает с именем класса;
- B) `init`;
- C) `_init_`;
- D) `__init__`

3. Рассмотрим вот такой фрагмент программы:

```
t = ([1], [2], 3, 10)
t1 = t[0]
t1.append(10)
```

Как лучше всего охарактеризовать этот фрагмент?

- A) Этот фрагмент содержит ошибку, так в нем есть попытка изменить значение переменной класса `tuple`;
- B) Корректный фрагмент кода;
- C) В этом фрагменте есть синтаксическая ошибка.

4. Какие из функций обязаны содержать объявление `global point` чтобы

изменить значение глобальной переменной point?

```
point = [0, 0]

def function1():
    point[0] += 1
    point[1] += 1

def function2():
    point = [50, 50]

def function3():
    point.append(30)
    point.pop(0)

def function4():
    point[0], point[1] = 30, 40
```

- A) function1 ();
- B) function2 ();
- C) function3 ();
- D) function4 ();

5. Рассмотрим следующую программу:

```
class Account:
    def __init__(self, id):
        self.id = id
        id = 666

acc = Account(123)
print(acc.id)
```

Что будет выведено на экран в результате работы следующей программы?

- A) 123;
- B) В коде есть синтаксическая ошибка, на экран ничего не будет выведено;
- C) 666;
- D) None

## Раздел «Реализация алгоритмов на языке Python»

### 26.14. Тема «Переборные алгоритмы»

1. Вы планируете найти самый выгодный рейс из Омска в Берлин. Однако эпидемия внесла свои коррективы и аэропорты закрыли вылеты в некоторые города.

Вы знаете, что существует пять рейсов из Омска в Новосибирск. По прилету каждым из таких рейсов, вы сможете воспользоваться семью рейсами из Новосибирска в Москву. После чего есть четыре рейса из Москвы в Берлин.

Сколько всего рейсов вам надо перебрать, чтобы проанализировать все возможные способы добраться до пункта назначения?

2. Вы решаете хитрую задачу, для которой написали программу, реализующую перебор:

```
for a in range(21):
    for b in range(21):
        for c in range(21):
            for d in range(c):
                for e in range(d):
                    for f in range(103):
                        for g in range(f):
                            for h in range(g):
                                print(a, b, c, d, e, f, g,
```

Сколько элементов вам придется перебрать?

3. На прямой располагаются точки, координата каждой точки целое число. Вам надо найти две ближайшие друг другу точки.

Станет ли более простым (с точки зрения реализации и количества перебираемых элементов) перебор, если вы потребуете, чтобы координаты точек задавались в порядке возрастания?

- A) Да;
- B) Нет.

4. Рассмотрим следующий фрагмент программы:

```
for i in range(5):
    for j in range(i):
        for k in range(j):
            # Используем набор (i, j, k)
```

Что здесь перебирается?

- A) перестановки;
- B) 3-сочетания без повторений;
- C) 3-размещения без повторений;
- D) 3-сочетания с повторениями;
- E) 3-размещения с повторениями.

5. Вы играете в покер со своим другом колодой в 52 карты. Сначала он сдает вам две карты, которые вы видите. Потом он выкладывает три карты на стол в

открытую. После чего сдает две карты себе рубашкой вверх.

Вы решили перебрать все варианты того, какие карты ему могут попасться чтобы проанализировать свои шансы. В покере порядок карт у игрока и на столе значения не имеет.

Сколько всего вариантов вам надо перебрать?

## 26.15. Тема «Переборные алгоритмы» - контеcт

### 1. Ферзи под ударом

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

На обычной шахматной доске 8x8 расставлены ферзи. Вам надо вывести все пары ферзей, которые бьют друг друга. Каждую такую пару надо вывести ровно один раз.

#### Формат ввода

В первой строке через пробел вводятся координаты ферзей в стандартной шахматной нотации. Гарантируется, что позиция ферзя является корректной, два ферзя не занимают одну и ту же клетку. Список ферзей непустой.

#### Формат вывода

Вывести все пары ферзей, которые бьют друг друга. Каждую пару ферзей надо выводить в отдельной строке через пробел. Порядок вывода (пар и ферзей в паре) значения не имеет. Координата каждого из ферзей приводится в стандартной шахматной нотации.

#### Пример

Ввод

Qa1 Qb3 Qc5 Qd2

Вывод

### 2. Рюкзак из трёх предметов

Ограничение времени	0.3 секунды
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Перед вами на ходятся несколько предметов, обладающих стоимостью и весом. Вам нужно определить максимальную стоимость набора из трёх предметов, которые можно положить в рюкзак, способный выдержать

выдержать предметы общим весом  
M.

Формат ввода

На вход подается три строки. В первой строке набор стоимостей предметов, во второй набор их весов. Количество предметов больше или равно трем и не превосходит 100. Третья строка содержит максимальный вес M, который способен выдержать рюкзак.

Формат вывода

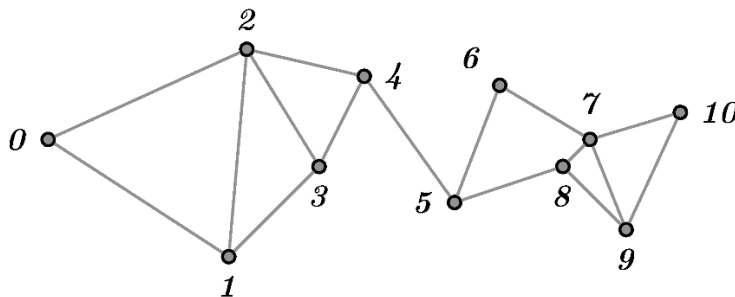
Число, показывающее максимальную стоимость рюкзака состоящего из трёх предметов. Если ни одного набора из трёх предметов не вмещается в рюкзак, вывести -1

Пример

Ввод	Вывод
1 3 2	6
2 4 4	
10	

## 26.16. Тема «Графовые алгоритмы»

1. Рассмотрим следующий граф:.



Допустим, что мы раскрашиваем вершины этого графа вот в таком порядке:

[10, 0, 2, 7, 8, 9, 3, 6, 5, 1, 4]

Цвета, в которые мы раскрашиваем вершины, обозначаются константами 0, 1, 2,...

В какой цвет будет раскрашена вершина 4?

2. Допустим у вас есть граф на 10 вершинах.

Сколько раз вам надо будет запустить алгоритм жадной раскраски в самом плохом случае, чтобы найти хроматическое число графа?

3. Выберите правильные утверждения, которые касаются понятия дерева.

А) произвольный граф можно преобразовать в дерево;

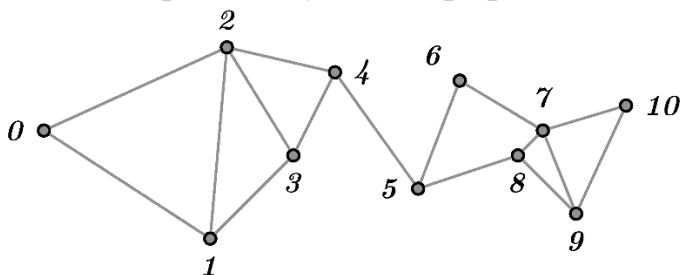
В) в дереве не может быть циклов;

С) любое дерево является связным графом;



D) в дереве могут быть две вершины, между которыми может быть несколько различных путей

4. Рассмотрим следующий граф:.

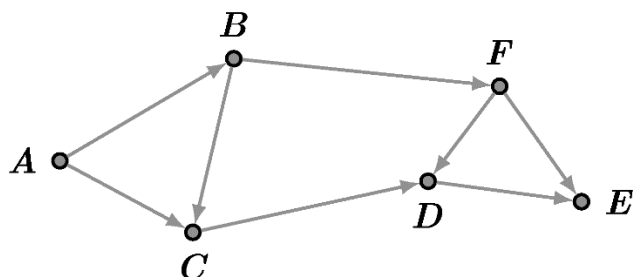


Этот граф может быть задан при помощи следующих списков смежности:

[[0, 2, 1], [0, 3, 2], [0, 4, 3, 1], [1, 2, 4], [2, 3, 5], [4, 6], [5, 7], [6, 8, 9, 10], [7, 9], [7, 8, 10], [7, 9]]

Приведите через пробел порядок, в котором произойдет обход вершин этого графа, если мы это делаем при помощи поиска в ширину начиная с вершины 0. Мы считаем, что вершина исследована в тот момент, когда она была извлечена из очереди to\_explore.

5. Рассмотрим следующий граф, моделирующий дорожную сеть. В этой сети можно двигаться только по направлению движения стрелок.



Сколько всего маршрутов в этом графе из вершины A до вершины E?

26.17. Тема «Графовые алгоритмы» - констест

1. Вторая марсианская конференция.

Ограничение времени	0.2 секунды
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

На Марсианские конференции собираются ученые со всего Марса. Для простоты общения марсианские ученые берут себе вместо имен числительные. При регистрации каждый участник конференции указывает список своих друзей, с которыми он хочет посетить одну секцию работы конференции. Если

это совсем молодой ученый (еще не успел завести себе друзей), то он при регистрации указывает пустой список.

Напишите программу, которая подсчитывает количество секций на конференции.

Формат ввода

Первая строка ввода содержит одно число  $N$  ( $N \leq 100$ ) - количество марсиан на конференции. Следующие  $N$  строк содержат списки друзей зарегистрированных марсиан от 1 до  $N$ . Помните, что отношение дружбы у марсиан всегда взаимное.

Формат вывода

Единственная строка вывода вашей программы должна содержать количество секций на конференции.

Пример

Ввод	Вывод
5	1
2	
1 3 4 5	
2	
2	
2	

## 2. Шесть шагов до Кевина Бэйкона.

Ограничение времени	60 секунд
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Шесть шагов до Кевина Бэйкона (англ. Six Degrees of Kevin Bacon) - игра, участники которой должны не более чем за 6 переходов найти связь между загаданным актёром и Кевином Бэйконом через актёров, вместе с которыми они снимались. Игра основана на теориях тесного мира и шести рукопожатий. Вам надо определить минимальное количество шагов от Кевина Бэйкона до заданного актёра. У самого Кевина Бэйкона это число равно 0.

Формат ввода

На вход подается имя актёра в виде строки. Также в одном каталоге с программой будет находиться файл movies.txt, в котором перечисляются фильмы вместе с актерским составом. В каждой строке представлены данные по одному фильму, например вот так:

Alien: Resurrection (1997)/Faltisco, Robert/Gilborn, Steven/St. James, David/Larioza, Nito/Ramessar, Ronald/Damer, Rod/Freeman, J.E./Lorre, Alex

(I)/Wincott, Michael/Mitchell, Rodney (I)/Bastens, Robert/Woodruff Jr., Tom/Orser, Leland/D'Annunzio, Cris/Raymont, Daniel/Yansick, Eddie/Bueno, Rico/Hahn, Archie/Perlman, Ron/Hedaya, Dan/Cruz, Raymond/Rowe, David (III)/House, Garrett/Prior, David Britten/Mansfield, Mark/Dourdan, Gary/Dourif, Brad/Martin, Brad (I)/Pinon, Dominique/Fellows, Nicole/Bush, Marlene/DuCasse, Regan/Campbell, Carolyn/LaBarbara, Joan/Flowers, Kim (I)/Ryder, Winona/Weaver, Sigourney

Первым идет название фильма, а потом через слэш перечисляются актеры.

Для нужд тестирования вы можете скачать этот файл отсюда - <https://yadi.sk/i/ICTmR4Wr3TeWct>

Формат вывода

Единственное число, которое задает минимальное количество шагов от Кевина Бэйкона до заданного актера.

В том случае если пути от Кевина Бэйкона до актера не существует, необходимо вывести слово "Infinity".

Если введенный актер не найден в файле movies.txt, то надо вывести "Unknown".

Пример

Ввод	Вывод
Beckham, Victoria	2

## 26.18. Тема «Алгоритмические задачи»

### 1. Самая выгодная сделка.

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод
Вывод	стандартный вывод

Главный принцип торговли на бирже: дешевле купить и подороже продать. Петр разрабатывает новую алгоритмическую торговую систему, в которой ему надо определить самую выгодную сделку за предыдущий период времени торговли финансового актива.

Сделка однократная, сначала актив один раз покупается и через какое-то время продается.

Если сделок с положительным финансовым результатом нет, то вывести 0.0

Формат ввода

В первой строке подается одно число  $2 < N \leq 100000$ .

Далее следует одна строка, в которой задается последовательность цен на актив в течение интересующего Петра периода времени длины  $N$  действительных чисел  $1.0 \leq a_i \leq 500.0$ .

Формат вывода

Надо вывести финансовый результат самой выгодной сделки с округлением до одного знака после запятой, либо 0.0, если такую сделку невозможно осуществить.

Пример

Ввод	Вывод
4	4.2
1.0 5.2 3.0 2.0	

## 2. Два алгоритма

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

В Центре исследований искусственного интеллекта решили научить искусственный интеллект играть в следующую игру. На столе в ряд лежат карточки с числами. Карточки берутся слева направо. На каждом ходу можно взять либо одну, либо ноль карточек. Один раз за всю игру можно взять со стола две карточки. Игра продолжается до тех пор, пока все карточки не будут взяты. Выигрывает тот, у кого сумма чисел на собранных карточках будет максимальной.

Исследователи решили, что сравнивать игроков будут с жадным алгоритмом, который на каждом своем ходу будет брать ровно одну карточку. Жадный алгоритм всегда ходит вторым.

Какую максимальную сумму может собрать правильно играющий против такого соперника алгоритм?

Формат ввода

В первой строке число  $N$  — количество карточек на столе. ( $N \leq 100000$ )

Во второй строке через пробел записаны  $N$  неотрицательных чисел, которые записаны на карточках. Каждое число не превышает 100.

Формат вывода

Одно число — максимальный результат, который получится у правильно играющего алгоритма.

Пример

Ввод

13

1 2 3 4 5 6 7 8 9 10 8 7 1

Вывод

42

### Модуль «Введение в базы данных»

#### 26.19. Тема «Введение в базы данных. Таблицы»

1. Что хранится в базе данных?
  - А) Данные.
  - В) Информация.
  - С) Метаданные.
2. Отметьте все свойства, характеризующие реляционную таблицу.
  - А) Записи в реляционной таблице уникальны.
  - В) Записи и столбцы упорядочены определённым образом.
  - С) Значения в ячейках элементарны.
  - Д) Первичный ключ, состоящий из нескольких столбцов, может содержать NULL-значение, но только одно.
3. Что из перечисленного не является суперключом?
  - А) Первичный ключ.
  - В) Потенциальный ключ.
  - С) Внешний ключ.
  - Д) Множество всех столбцов таблицы.

#### 26.20. Тема «Запросы на выборку данных»

1. Дана таблица Student следующей структуры:  
*Student* (id\*, name, AGroup, agrant).  
Первичный ключ отмечен звёздочкой.  
Напишите запрос для выборки всех студентов, получающих стипендию.
2. Дана таблица Student следующей структуры:  
*Student* (id\*, name, AGroup, agrant).  
Первичный ключ отмечен звёздочкой.  
Напишите запрос, выводящий список названий групп с указанием средней стипендии, посчитанной только по студентам, получающим её.
3. Дана таблица AGroup следующей структуры:  
*AGroup* (id\*, course, department, head).  
Первичный ключ отмечен звёздочкой.  
Напишите запрос для выборки всех групп, где не назначен староста (head).
4. Дана таблица Concert следующей структуры:  
*Concert* (name\*, date\_c\*, tickets\_sold, ticket\_price, singer\_name).  
Первичный ключ отмечен звёздочкой.

Напишите запрос для выборки всех сегодняшних концертов, на которые не продано ни одного билета.

#### 26.21. Тема «Запросы на изменение данных»

1. Какие операции можно откатить (в пределах транзакции) с помощью команды ROLLBACK?

- A) UPDATE.
- B) DELETE.
- C) SELECT.
- D) INSERT.
- E) TRUNCATE.

2. Какие ограничения можно определять сразу для нескольких столбцов?

- A) PRIMARY KEY.
- B) FOREIGN KEY.
- C) UNIQUE.
- D) NOT NULL.
- E) CHECK.

3. Дана таблица Student следующей структуры:

*Student* (id\*, name, AGroup, agrant).

Первичный ключ отмечен звёздочкой.

Напишите запрос для добавления в таблицу *AGroup* записи о группе с названием «ИТ-21МО», курсом 6 и факультетом «ИВТ».

4. Дана таблица Student следующей структуры:

*Student* (id\*, name, AGroup, agrant).

Первичный ключ отмечен звёздочкой.

Напишите запрос для удаления из таблицы *Student* всех студентов по фамилии «Петров», не получающих стипендии.

5. Дана таблица Student следующей структуры:

*Student* (id\*, name, AGroup, agrant).

Первичный ключ отмечен звёздочкой.

Напишите запрос для изменения номера группы на NULL для студентов тех групп, численность которых меньше 10 студентов.

6. Дана таблица Concert следующей структуры:

*Concert* (name\*, date\_c\*, tickets\_sold, ticket\_price, singer\_name).

Первичный ключ отмечен звёздочкой.

Напишите запрос, дублирующий в таблице *Concert* записи о тех концертах, на которые продано больше 5000 билетов. При этом в названии концерта указывается суффикс «-2», а дата переносится на месяц вперёд. Количество проданных билетов на добавленный концерт равно 0.

#### 26.22. Тема «Сложные запросы с использованием нескольких таблиц»

1. Дан SQL-запрос:

```

SELECT *
FROM Student
WHERE agrant = (
    SELECT agrant
    FROM Student
    WHERE surname = 'Иванов'
);

```

В каких случаях этот запрос будет работать корректно?

- A) Если в таблице нет ни одного студента по фамилии Иванов.
- B) Если в таблице есть ровно один студент по фамилии Иванов.
- C) Если в таблице несколько студентов по фамилии Иванов, но у всех стипендия одинаковая.
- D) Если в таблице несколько студентов по фамилии Иванов с разными стипендиями.

2. Для каких типов соединения таблиц в ветке FROM запроса SELECT важен порядок указания участвующих в этом соединении таблиц?

- A) Регулярное соединение.
- B) Естественное соединение.
- C) Левое внешнее соединение.
- D) Правое внешнее соединение.
- E) Полное внешнее соединение.

3. Какие операции над множествами могут привести к тому, что итоговая выборка будет содержать дубликаты значений?

- A) UNION.
- B) UNION ALL.
- C) INTERSECT.
- D) MINUS.

4. База данных содержит информацию о сотрудниках и отделах, которая представлена двумя таблицами:

Таблица **Employees**

emp_no*	name	salary	dept_name
1	Романов	40 000 р.	Бухгалтерия
2	Васильев	35 000 р.	ИТ
3	Кириллов	47 500 р.	ИТ
4	Петров	39 000 р.	ИТ
5	Петров	29 000 р.	Бухгалтерия
6	Сидоров	19 000 р.	Бухгалтерия

Таблица **Departments**

dept_name*	manager_id
ИТ	3
Бухгалтерия	1

Здесь знаком «\*» обозначены первичные ключи; начальник отдела является его работником, при этом допускается пустое значение данного поля.

Требуется определить, какое значение будет возвращено запросом:

```
SELECT ROUND(AVG(salary), -3)
FROM Employees
WHERE (name NOT LIKE 'П%' OR dept_name <> 'ИТ')
AND emp_no NOT IN (
    SELECT manager_id
    FROM Departments);
```

В ответе укажите целое число без пробелов и буквенных обозначений «р.».

5. База данных содержит информацию о сотрудниках и отделах, которая представлена двумя таблицами:

Таблица **Employees**

emp_no*	name	salary	dept_name
1	Романов	40 000 р.	Бухгалтерия
2	Васильев	35 000 р.	ИТ
3	Кириллов	47 500 р.	ИТ
4	Петров	39 000 р.	ИТ
5	Петров	29 000 р.	Бухгалтерия
6	Сидоров	19 000 р.	Бухгалтерия

Таблица **Departments**

dept_name*	manager_id
ИТ	3
Бухгалтерия	1

Здесь знаком «\*» обозначены первичные ключи; начальник отдела является его работником, при этом допускается пустое значение данного поля.

Требуется определить, какое значение будет возвращено запросом:

```
SELECT (MAX(salary) – MIN(salary))
FROM Employees
WHERE emp_no NOT IN (
    SELECT manager_id
    FROM Departments)
OR salary <= 40000;
```

В ответе укажите целое число без пробелов и буквенных обозначений «р.».

6. База данных состоит из двух таблиц следующей структуры:

- *Student* (id\*, name, AGroup, agrant);
- *AGroup* (id\*, course, department, head).

Первичные ключи отмечены звёздочкой, внешние – подчёркиванием. Староста (head) должен быть студентом соответствующей группы.

Напишите запрос, выводящий список групп с указанием факультета и имени старосты, где староста получает стипендию большую, чем стипендия каждого



из его одногруппников.

7. База данных состоит из двух таблиц следующей структуры:

- *Student* (id\*, name, AGroup, agrant);
- *AGroup* (id\*, course, department, head).

Первичные ключи отмечены звёздочкой, внешние – подчёркиванием.

Староста (head) должен быть студентом соответствующей группы.

Напишите запрос, выводящий список студентов 1 курса, получающих стипендию большую, чем любой студент 4 курса того же факультета.

8. База данных состоит из двух таблиц следующей структуры:

- *Singer* (name\*, country, royalties, birthday);
- *Concert* (name\*, date\_c\*, tickets\_sold, ticket\_price, singer\_name).

Первичные ключи отмечены звёздочкой, внешние – подчёркиванием.

Староста (head) должен быть студентом соответствующей группы.

Напишите запрос, выводящий список таких стран, что ни один из исполнителей этих стран не выступал в последнюю неделю.

## 26.23. Тема «Проектирование баз данных»

1. Какие типы связей в ER-модели требуют создания дополнительных сущностей при переходе к реляционной модели?

- A) Бинарная связь 1:M.
- B) Бинарная связь N:M.
- C) Унарная связь 1:M.
- D) Унарная связь N:M.
- E) Связь, имеющая атрибуты.

2. Отметьте все правильные утверждения относительно преобразования сущностей ER-модели в таблицы базы данных.

- A) Каждая сущность ER-модели представляется в виде таблицы базы данных.
- B) Каждая сущность ER-модели представляется в виде столбца таблицы базы данных.
- C) Экземпляру сущности соответствует столбец таблицы базы данных.
- D) Экземпляру сущности соответствует запись в таблице базы данных.

3. Отметьте все свойства, характеризующие вторую нормальную форму реляционной таблицы.

- A) Ни одно поле не содержит более одного значения.
- B) Ни одно из ключевых полей не пусто.
- C) Каждый столбец, не входящих в состав первичного ключа, функционально полно зависит от этого ключа.
- D) Нет ни одного не входящего в первичный ключ столбца, находящегося в транзитивной функциональной зависимости от ключа.

4. Дана схема реляционной таблицы «Билет»:

{Номер рейса\*, Номер места\*, ФИО пассажира, Стоимость билета, Аэропорт назначения, Город назначения}.

Звёздочкой обозначен первичный ключ. Какие столбцы находятся в полной функциональной зависимости от первичного ключа?

- A) ФИО пассажира.
- B) Стоимость билета.
- C) Аэропорт назначения.
- D) Город назначения.

5. Дана схема реляционной таблицы «Билет»:

{Номер рейса\*, Номер места\*, ФИО пассажира, Стоимость билета, Аэропорт назначения, Город назначения}.

Звёздочкой обозначен первичный ключ. Какие столбцы находятся в транзитивной зависимости от первичного ключа?

- A) ФИО пассажира.
- B) Стоимость билета.
- C) Аэропорт назначения.
- D) Город назначения.

## 26.24. Тема «Введение в ORM»

1. Что такое ORM?

- A) Object-Relational Memory.
- B) Object-Relational Market.
- C) Object-Relational Mapping.
- D) Object-Relational Model.

2. Что такое QuerySet?

- A) С помощью QuerySet можно отправлять запросы.
- B) Это набор каких-то запросов.
- C) Это СУБД.
- D) Это список объектов заданной модели.

3. Для чего используется QuerySet?

- A) QuerySet позволяет читать данные из базы данных, фильтровать и изменять их порядок.
- B) QuerySet позволяет создавать базы данных.
- C) С помощью QuerySet можно редактировать содержимое базы данных.

4. Что делает метод all()?

- A) Возвращает копию текущего QuerySet.
- B) Возвращает все элементы из QuerySet.
- C) Удаляет все данные из QuerySet.
- D) Копирует все данные из QuerySet.

5. Что делает метод union()?

- A) Использует оператор SQL UNION для разъединения результатов двух

или более QuerySet'ов.

В) Использует оператор SQL UNION для удаления результатов двух или более QuerySet'ов.

С) Использует оператор SQL UNION для редактирования результатов двух или более QuerySet'ов.

Д) Использует оператор SQL UNION для объединения результатов двух или более QuerySet'ов.

6. Что делает метод intersection()?

А) Использует оператор SQL INTERSECT для возврата различных элементов двух или более QuerySet'ов.

В) Использует оператор SQL INTERSECT для удаления различных элементов двух или более QuerySet'ов.

С) Использует оператор SQL INTERSECT для возврата общих элементов двух или более QuerySet'ов.

Д) Использует оператор SQL INTERSECT для удаления общих элементов двух или более QuerySet'ов.

7. Что делает метод difference()?

А) Использует оператор SQL EXCEPT для хранения только элементов, отсутствующих в QuerySet.

В) Использует оператор SQL EXCEPT для добавления элементов, отсутствующих в QuerySet.

С) Использует оператор SQL EXCEPT для хранения только элементов, присутствующих в QuerySet, но не в каких-либо других QuerySet'ах.

Д) Использует оператор SQL EXCEPT для удаления элементов, присутствующих в QuerySet.

8. Что делает метод using()?

А) Этот метод предназначен для указания, какую базу данных QuerySet будет использовать, если вы используете более одной базы данных.

В) Этот метод предназначен для выбора базы данных, если вы используете более одной базы данных.

С) Этот метод возвращает все используемые базы данных.

Д) Этот метод предназначен для указания, какую базу данных QuerySet не будет использовать, если вы используете более одной базы данных.

## **27. Промежуточная аттестация. Перечень примерных заданий**

### **Раздел «Введение в информатику и программирование»**

#### **27.1. Тема «Итоговая аттестация по разделу»**

1. Сколькими символами (цифрами) записываются троичные значения:

А) 8;

В) 4;

С) 3.

2. Бит может принимать:

- А) два значения: 0 и 1
- В) значения в диапазоне от 0 до 255
- С) значения в диапазоне от -127 до 128

3. Стандарт ASCII касается:

- А) кодирования чисел;
- В) кодирования текста;
- С) использования носителей информации (жесткие диски, флешки и прочие).

4. Байт состоит из:

- А) четырёх бит;
- В) восьми бит;
- С) шестнадцати бит.

5. Первым компьютером общего назначения считают:

- А) ENIAC;
- В) Harvard Mark I;
- С) IBM PC.

## Раздел «Основы программирования на языке Python»

### 27.2. Тема «Итоговая аттестация по разделу»

1. Интересный делитель.

Ограничение времени	1 секунда
Ограничение памяти	256Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Интересным делителем называется число  $N$ , которое является делителем числа  $N+1$ . Вам требуется найти количество интересных делителей на отрезке от  $L$  до  $R$ , концы отрезка включаются.

Формат ввода

Во входном файле записано через пробел два числа —  $L$  и  $R$  ( $L < R$ ,  $1 \leq L, R \leq 10^{18}$ ).

Формат вывода

Вывести единственное число — ответ на задачу.

Пример

Ввод	Вывод
2 5	0

## 2. Четные, но не кратные шести.

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.tx

Рассмотрим массив  $A$  размера  $N$ , в котором хранятся целые числа. Найдите количество подпоследовательностей длины  $K$  в этом массиве, которые состоят только из четных чисел, каждое из которых не кратно 6. Подпоследовательность формируется подряд идущими элементами массива.

### Формат ввода

В первой строке подаются два числа  $2 < N \leq 300000$  и  $1 \leq K \leq N$ . За ними следует одна строка, в которой перечислены  $N$  чисел массива  $A$  через пробел. Все числа неотрицательные и не превосходят 1000.

### Формат вывода

Одно целое число, задающее количество искомых подпоследовательностей в массиве  $A$ .

### Пример

Ввод	Вывод
6 3	1
2 4 6 2 4 8	

### Примечания

0 мы считаем четным числом, кратным 6.

Если вы решите задачу для  $K=3$ , то сможете заработать максимум 10 баллов;

- решение задачи для произвольного  $K$  принесет вам максимум 20 баллов;
- эффективный алгоритм решения задачи для произвольного  $K$  позволит вам заработать 25 баллов

## Раздел «Реализация алгоритмов на языке Python»

### 27.3. Тема «Итоговая аттестация по разделу»

#### 1. Атакующий футбол.

Ограничение времени	0.15 секунд
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Ваша мечта сбылась - вы стали тренером футбольной команды и вам предстоит матч, который определит, останетесь ли вы в высшей лиге или нет.

Поскольку ближайший соперник плохо играет в защите и игра проходит у вас дома, то вы решаете играть с акцентом на атаку, максимально усилив созидательную линию своей команды.

Чтобы реализовать этот план вы посмотрели свежим взглядом на свою команду (включая дубль, фарм-клуб и молодежку) и оценили каждого игрока по тому, насколько хорошо он умеет играть в полузащите и нападении. Каждая оценка была выставлена по шкале от 0 до 100.

Для полузащитника ключевым умением является игра в полузащите, для нападающего ключевым является игра в нападении.

Вам надо выбрать четырех полузащитников и двух нападающих, сумма ключевых навыков которых является максимальной.

#### Формат ввода

В первой строке на вход подается число  $N$  ( $11 \leq N < 1000$ ), которое задает количество игроков. В каждой из следующих  $N$  строк через пробел записано два целых числа от 0 до 100, которые задают умение игрока играть в полузащите и нападении.

#### Формат вывода

Выводится одно число, которое задает сумму ключевых умений выбранных игроков.

#### Пример

Ввод	Вывод
16	552
22 5	
80 54	
63 77	
9 87	
93 31	
64 71	
47 3	
12 69	
16 32	
86 54	
36 48	
53 62	
95 41	
100 61	
19 0	
100 14	

## 2. Несуразные отношения.

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод

Вывод	стандартный вывод
-------	-------------------

*Судьба не соединяет равных...*

*Фольклор.*

Так считает директор только что открывшегося брачного агентства "Паросочетание" Пётр. По мнению Петра, первой парой, которую их агентству предстоит свести, должна быть союзом противоположностей.

Всего на данный момент в агентство анкеты подали  $N$  женщин и  $M$  мужчин. Каждому из них, согласуясь с написанным в анкете, Пётр смог назначить соответственно уровень шумности -  $a_i$  для женщин и  $b_i$  для мужчин.

По мнению директора, самым успешным сочетанием будет пара, в которой женщина имеет наибольший из всех кандидаток уровень шумности, а мужчина, в свою очередь, самый спокойный из имеющихся кандидатов.

Пётр хочет узнать уровень шумности получившейся пары и поэтому обратился к вам за помощью.

Формат ввода

В первой строке содержится  $N$  попарно различных чисел  $a_i$  ( $1 \leq a_i \leq 10^9$ ) - уровни шумности женщин.

Во второй строке содержится  $M$  попарно различных чисел  $b_i$  ( $1 \leq b_i \leq 10^9$ ) - уровни шумности мужчин.

Формат вывода

В единственной строке вывести число - суммарный уровень шумности искомой пары.

Пример

Ввод

1 3 2

5 1 2 4 6 3

Вывод

4

## Раздел «Введение в базы данных»

### 27.4. Тема «Итоговая аттестация по разделу»

1. База данных состоит из двух таблиц следующей структуры:

– *Lector* ( $id^*$ , name, birthday, degree);

– *Discipline* ( $cipher^*$ , name, hours, lector\_id, elaborator).

Первичные ключи отмечены звёздочкой, внешние – подчёркиванием. Разработчик курса (elaborator) является лектором, но может данную дисциплину не вести.

А) Напишите запрос для выборки всех дисциплин, где не назначен лектор.

В) Напишите запрос, выводящий список лекторов с указанием учёной степени и суммарной нагрузки, сгруппированный по лекторам и учёной степени, отдельно по степеням и по университету в целом.

С) Напишите запрос, выводящий список дисциплин, разработчик которых старше лектора.

Д) Напишите запрос для добавления лектора без степени с именем «Петров», идентификатором 32 и датой рождения 21.12.1980.

Е) Напишите запрос для увеличения в полтора раза количества часов для дисциплин, где лектор и разработчик имеют учёную степень, причём одинаковую.

## **28. Практика.**

Практика нацелена на приобретение конкретных практических навыков в области создания алгоритмов и программ, пригодных для практического применения, а также опыта самостоятельной профессиональной деятельности. Форма проведения практики – выполнение производственного задания и самостоятельная работа над ним. За время практики обучающийся должен получить задание, наметить пути решения, а также создать прототип решения.

Пример задания:

В компании разработан фронтенд простого решения для компании-брокера, которое позволяет проводить учет портфелей клиентов, отображать статистику по их доходности и проводить простую аналитику по времени совершения самой выгодной сделки (т.е. сделки между минимальной ценой покупки и максимальной ценой продажи актива). Интерфейс между фронтендом и бэкендом также уже реализован.

Вам надо реализовать:

- БД для хранения данных о клиентах и их портфелях;
- Данные по цене отдельных активов уже записаны в файловой системе в виде csv-файлов
- Алгоритм формирования стоимости портфеля в заданный промежуток времени
- Алгоритм нахождения дат, между которыми совершается самая выгодная сделка.
- Функции для изменения данных о клиентах, а также их портфелях.

В рамках практики требуется спроектировать базу данных и реализовать прототип решения.

По результатам практики представляется отчет о проделанной работе.

## **29. Итоговая аттестация.**

В рамках подготовки к защите требуется доработать проект, полученный на практике и представить его к защите.

Проект выполняется на базе теоретических знаний, практических умений, навыков и опыта деятельности, приобретенных студентом в период освоения ДПП ПП.

На основе проведенных работ делается доклад, где отражается объект, цели и



задачи работы, дается краткая характеристика содержания работы, показываются достигнутые результаты, подчеркивается новизна и авторский подход. Длительность доклада примерно 10 минут. Доклад сопровождается презентацией.

Степень сформированности компетенций выпускника оценивает комиссия в процессе защиты работы на основе анализа качества выполнения этапов:

- постановка задачи и ее решение;
- подготовка к защите и защита работы.

При оценке уровня профессиональной подготовленности по результатам защиты необходимо учитывать следующие основные параметры: масштабность работы; реальность поставленных задач; новизну предложенных технологических решений; качество доклада; качество и полноту ответов на вопросы.

## **ХII. Материально-техническое и учебно-методическое обеспечение Программы**

Для осуществления образовательного процесса используются средства платформы DemidOnline (<https://demidonline.uniyar.ac.ru/>) и технические средства для проведения онлайн-конференций.

Для выполнения практических заданий обучающимися требуется установка следующего свободно/бесплатно распространяемого ПО:

- язык программирования Python 3
- среда разработки PyCharm Community Edition
- СУБД MySQL.
- средства для администрирования СУБД MySQL Workbench Community Edition.

Для подготовки отчетов по практике и презентаций во время итоговой аттестации используется офисный пакет LibreOffice.

## **ХIII. Список литературы**

1. Доусон М. Программируем на Python. СПб.: Питер, 2015. – 416 с.
2. Саммерфильд М. Python на практике. ДМК Пресс, 2014. – 338 с.
3. Лутц М. Python: карманный справочник. Вильямс, 2015. – 320 с.
4. Кнут Д. Э. Искусство программирования. Том 3. Сортировка и поиск. Вильямс, 2012. – 824 с.
5. Кнут Д. Э. Искусство программирования. Том 1. Основные алгоритмы. Вильямс, 2015 г. – 720 с.
6. Кнут Д. Э. Искусство программирования. Том 4А. Комбинаторные алгоритмы. Вильямс, 2015 г. – 960 с.
7. Дейт К. Дж. Введение в системы баз данных, 8-е издание: пер с англ. - М.: Издательский дом «Вильямс», 2008. – 1327 с.
8. Чалый Д.Ю., Смирнов А.В. Практическое использование баз данных в управлении. Ярославль, ЯрГУ, 2012 г.
9. Гордеев А.В., Молчанов А.Ю. Системное программное обеспечение: учебник для вузов. - СПб.: Питер, 2001.-736с.

### Авторы программы

№ п/п	ФИО	Ученая степень и ученое звание	Основное место работы, должность
1.	Чалый Д.Ю.	к.ф.-м.н., доцент	ЯрГУ им. П.Г. Демидова, декан факультета ИВТ
2.	Апальков И.В.	к.т.н.	ООО «А-Вижн», директор
3.	Смирнов А.В.	к.ф.-м.н., доцент	ЯрГУ им. П.Г. Демидова, доцент каф. теоретической информатики
4.	Горбунов О.Е.	к.ф.-м.н.	ООО «Центр инновационного программирования», генеральный директор