

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Комарова Светлана Юриевна

Должность: Проректор по образовательной деятельности

Дата подписания: 06.07.2026 07:42:47

Уникальный программный ключ

43ba42f5deae4116bbfcb9ac98e39108031227e81add207cbee4149f2098d7a

**Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Омский государственный аграрный университет имени П.А. Столыпина»
Экономический факультет**

ОПОП по направлению 19.04.01 Биотехнология

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

по освоению учебной дисциплины

Б1.О.09 Алгоритмы и структуры больших данных

Направленность (профиль)

«Биоинформатика и анализ данных в агробиотехнологиях»

Обеспечивающая преподавание дисциплины кафедра	Экономики, бухгалтерского учета и финансового контроля
Разработчик, канд. экон. наук, доцент	А.А. Ремизова

СОДЕРЖАНИЕ

Введение

1. Место учебной дисциплины в подготовке
 2. Структура учебной работы, содержание и трудоёмкость основных элементов дисциплины
 - 2.1. Организационная структура, трудоемкость и план изучения дисциплины
 - 2.2. Содержание дисциплины по разделам
 3. Общие организационные требования к учебной работе обучающегося, условия допуска к экзамену
 - 3.1. Организация занятий и требования к учебной работе обучающегося
 - 3.2. Условия допуска к экзамену по дисциплине
 4. Лекционные занятия
 5. Практические занятия по курсу и подготовка обучающегося к ним
 6. Общие методические рекомендации по изучению отдельных разделов дисциплины
 7. Общие методические рекомендации по оформлению и выполнению отдельных видов ВАРС
 - 7.1. Рекомендации по написанию отчета
 - 7.1.1. Шкала и критерии оценивания
 - 7.2. Рекомендации по самостоятельному изучению тем
 - 7.2.1. Шкала и критерии оценивания
 8. Текущий (внутрисеместровый) контроль хода и результатов учебной работы обучающегося
 9. Промежуточная (семестровая) аттестация
 - 9.1. Нормативная база проведения промежуточной аттестации по результатам изучения дисциплины
 - 9.2. Основные характеристики промежуточной аттестации по итогам изучения дисциплины
 - 9.3. Подготовка к заключительному тестированию по итогам изучения дисциплины
 - 9.3.1. Шкала и критерии оценивания
 10. Учебно-информационные источники для изучения дисциплины
- Приложение 1 Форма титульного листа отчета
- Приложение 2 Результаты проверки отчета

ВВЕДЕНИЕ

1. Настоящее издание является основным организационно-методическим документом учебно-методического комплекса по дисциплине в составе основной профессиональной образовательной программы высшего образования (ОПОП ВО). Оно предназначено стать для них методической основой по освоению данной дисциплины.

2. Содержательной основой для разработки настоящих методических указаний послужила Рабочая программа дисциплины, утвержденная в установленном порядке.

3. Методические аспекты развиты в учебно-методической литературе и других разработках, входящих в состав УМК по данной дисциплине.

4. Доступ обучающихся к электронной версии Методических указаний по изучению дисциплины, обеспечен в электронной информационно-образовательной среде университета.

При этом в электронную версию могут быть внесены текущие изменения и дополнения, направленные на повышение качества настоящих методических указаний.

Уважаемые обучающиеся!

Приступая к изучению новой для Вас учебной дисциплины, начните с вдумчивого прочтения разработанных для Вас кафедрой специальных методических указаний. Это поможет Вам вовремя понять и правильно оценить ее роль в Вашем образовании.

Ознакомившись с организационными требованиями кафедры по этой дисциплине и соизмерив с ними свои силы, Вы сможете сделать осознанный выбор собственной тактики и стратегии учебной деятельности, уберечь самих себя от неразумных решений по отношению к ней в начале семестра, а не тогда, когда уже станет поздно. Используя эти указания, Вы без дополнительных осложнений подойдете к промежуточной аттестации по этой дисциплине. Успешность аттестации зависит, прежде всего, от Вас. Ее залог – ритмичная, целенаправленная, вдумчивая учебная работа, в целях обеспечения которой и разработаны эти методические указания.

1. Место учебной дисциплины в подготовке выпускника

Учебная дисциплина относится к дисциплинам ОПОП университета, состав которых определяется вузом и требованиями ФГОС.

Цель дисциплины – получение магистрантами основополагающих знаний и практических навыков в области разработки алгоритмов и структур данных для обработки, анализа и интерпретации больших массивов биологической информации в агробиотехнологиях, с использованием языка программирования Python и специализированных библиотек для работы с большими объемами информации.

В ходе освоения дисциплины обучающийся должен:

иметь целостное представление о алгоритмах для решения вычислительных задач биоинформатики, основанных на больших данных;

владеть:

- методологией построения эффективных алгоритмических решений для обработки больших данных в агробиоинформатике;
- навыками работы с основными форматами NGS-данных и инструментами их первичной обработки и статистического анализа
- опытом использования высокопроизводительных и облачных вычислительных ресурсов для запуска ресурсоёмких пайплайнов;
- приёмами оптимизации кода применительно к биологическим задачам;
- способностью самостоятельно проектировать вычислительный эксперимент, оценивать его воспроизводимость и проверять статистическую значимость результатов на больших выборках;

знать:

- теоретические основы классических и современных алгоритмов биоинформатики;
- ключевые структуры данных для работы с большими геномными, транскриптомными и метагеномными данными;
- алгоритмические подходы к обработке данных NGS и распределённым вычислениям;
- специфику агробиотехнологических данных – генотипирование сельскохозяйственных растений и животных;
- принципы оценки вычислительной сложности и потребления памяти для выбора алгоритма под масштаб реальных задач;

уметь:

- разрабатывать и адаптировать формальные алгоритмы для решения типовых вычислительных задач биоинформатики в агросфере;
- выбирать адекватные структуры данных и индексы для эффективной обработки больших файлов;
- реализовывать на Python вычислительные конвейеры для анализа геномных и транскриптомных данных;
- применять современные программные пакеты и интегрировать их в скрипты

1.1.Перечень компетенций с указанием этапов их формирования в результате освоения учебной дисциплины:

Компетенции, в формировании которых задействована дисциплина		Код и наименование индикатора достижений компетенции	Компоненты компетенций, формируемые в рамках данной дисциплины (как ожидаемый результат ее освоения)		
код	наименование		знать и понимать	уметь делать (действовать)	владеть навыками (иметь навыки)
Общепрофессиональные компетенции					
ОПК-3	Способен разрабатывать алгоритмы и участвовать в разработке программ в сфере своей профессиональной деятельности	ИД-1 _{опк-3} Разрабатывает формальные алгоритмы для решения вычислительных задач биоинформатики, основанных на больших данных	Знает и понимает теоретические основы классических и современных алгоритмов биоинформатики;	Умеет разрабатывать и адаптировать формальные алгоритмы для решения типовых вычислительных задач биоинформатики в агросфере;	Владеет методологией построения эффективных алгоритмических решений для обработки больших данных в агробиоинформатике;
			ключевые структуры данных для работы с большими геномными, транскриптомным и и метагеномными данными;	выбирать адекватные структуры данных и индексы для эффективной обработки больших файлов;	навыками работы с основными форматами NGS-данных и инструментами их первичной обработки и статистического анализа опытом использования высокопроизводительных и облачных вычислительных ресурсов для запуска ресурсоёмких пайплайнов;
			алгоритмические подходы к обработке данных NGS и распределённым вычислениям;	реализовывать на Python вычислительные конвейеры для анализа геномных и транскриптомных данных;	приёмами оптимизации кода применительно к биологическим задачам;
			специфику агробиотехнологических данных – генотипирование сельскохозяйственных растений и животных;	применять современные программные пакеты и интегрировать их в скрипты	способностью самостоятельно проектировать вычислительный эксперимент, оценивать его воспроизводимость и проверять статистическую значимость результатов на больших выборках
			принципы оценки вычислительной сложности и потребления памяти для выбора алгоритма под масштаб реальных задач		

1.2. Описание показателей, критериев и шкал оценивания и этапов формирования компетенций в рамках дисциплины

Индекс и название компетенции	Код индикатора достижений компетенции	Индикаторы компетенции	Показатель оценивания – знания, умения, навыки (владения)	Уровни сформированности компетенций		Формы и средства контроля формирования компетенций		
				компетенция не сформирована	минимальный		средний	высокий
				Оценки сформированности компетенций				
				Не зачтено			Зачтено	
				Характеристика сформированности компетенции				
				Компетенция в полной мере не сформирована. Имеющихся знаний, умений и навыков недостаточно для решения практических (профессиональных) задач	1. Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач. 2. Сформированность компетенции в целом соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения стандартных практических (профессиональных) задач. 3. Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач.			
Критерии оценивания								
ОПК-3 Способен разрабатывать алгоритмы и участвовать в разработке программ в сфере своей профессиональной деятельности	ИД-1 _{ОПК-3} Разрабатывает формальные алгоритмы для решения вычислительных задач биоинформатики, основанных на больших данных	Полнота знаний	Знает и понимает теоретические основы классических и современных алгоритмов биоинформатики; ключевые структуры данных для работы с большими геномными, транскриптомными и метагеномными данными; алгоритмические подходы к обработке данных NGS и распределённым вычислениям; специфику агробиотехнологических данных – генотипирование сельскохозяйственных растений и животных; принципы оценки вычислительной сложности и потребления памяти для выбора алгоритма под масштаб реальных задач	Не знает: – теоретические основы классических и современных алгоритмов биоинформатики; – ключевые структуры данных для работы с большими геномными, транскриптомными и метагеномными данными; – алгоритмические подходы к обработке данных NGS и распределённым вычислениям; – специфику агробиотехнологических данных – генотипирование сельскохозяйственных растений и животных; – принципы оценки вычислительной сложности и потребления памяти для выбора алгоритма под масштаб реальных задач	Знает (на минимальном уровне, хорошо или отлично): – теоретические основы классических и современных алгоритмов биоинформатики; – ключевые структуры данных для работы с большими геномными, транскриптомными и метагеномными данными; – алгоритмические подходы к обработке данных NGS и распределённым вычислениям; – специфику агробиотехнологических данных – генотипирование сельскохозяйственных растений и животных; принципы оценки вычислительной сложности и потребления памяти для выбора алгоритма под масштаб реальных задач	Итоговое тестирование, опрос, проверка выполненных работ, отчет		

Индекс и название компетенции	Код индикатора достижений компетенции	Индикаторы компетенции	Показатель оценивания – знания, умения, навыки (владения)	Уровни сформированности компетенций		Формы и средства контроля формирования компетенций		
				компетенция не сформирована	минимальный		средний	высокий
				Оценки сформированности компетенций				
				Не зачтено			Зачтено	
				Характеристика сформированности компетенции				
				Компетенция в полной мере не сформирована. Имеющихся знаний, умений и навыков недостаточно для решения практических (профессиональных) задач	1. Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач. 2. Сформированность компетенции в целом соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения стандартных практических (профессиональных) задач. 3. Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач.			
Критерии оценивания								
ОПК-3 Способен разрабатывать алгоритмы и участвовать в разработке программ в сфере своей профессиональной деятельности	ИД-1 _{ОПК-3} Разрабатывает формальные алгоритмы для решения вычислительных задач биоинформатики, основанных на больших данных	Наличие умений	Умеет разрабатывать и адаптировать формальные алгоритмы для решения типовых вычислительных задач биоинформатики в агросфере;	Не умеет: – разрабатывать и адаптировать формальные алгоритмы для решения типовых вычислительных задач биоинформатики в агросфере; – выбирать адекватные структуры данных и индексы для эффективной обработки больших файлов; – реализовывать на Python вычислительные конвейеры для анализа геномных и транскриптомных данных; – применять современные программные пакеты и интегрировать их в скрипты	Умеет (на минимальном уровне, хорошо или отлично): – разрабатывать и адаптировать формальные алгоритмы для решения типовых вычислительных задач биоинформатики в агросфере; – выбирать адекватные структуры данных и индексы для эффективной обработки больших файлов; – реализовывать на Python вычислительные конвейеры для анализа геномных и транскриптомных данных; применять современные программные пакеты и интегрировать их в скрипты	Итоговое тестирование, опрос, проверка выполненных работ, отчет		
			выбирать адекватные структуры данных и индексы для эффективной обработки больших файлов;					
			реализовывать на Python вычислительные конвейеры для анализа геномных и транскриптомных данных;					
			применять современные программные пакеты и интегрировать их в скрипты					

Индекс и название компетенции	Код индикатора достижений компетенции	Индикаторы компетенции	Показатель оценивания – знания, умения, навыки (владения)	Уровни сформированности компетенций		Формы и средства контроля формирования компетенций		
				компетенция не сформирована	минимальный		средний	высокий
				Оценки сформированности компетенций				
				Не зачтено			Зачтено	
				Характеристика сформированности компетенции				
				Компетенция в полной мере не сформирована. Имеющихся знаний, умений и навыков недостаточно для решения практических (профессиональных) задач	1. Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач. 2. Сформированность компетенции в целом соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в целом достаточно для решения стандартных практических (профессиональных) задач. 3. Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и мотивации в полной мере достаточно для решения сложных практических (профессиональных) задач.			
Критерии оценивания								
ОПК-3 Способен разрабатывать алгоритмы и участвовать в разработке программ в сфере своей профессиональной деятельности	ИД-1 _{ОПК-3} Разрабатывает формальные алгоритмы для решения вычислительных задач биоинформатики, основанных на больших данных	Наличие навыков (владение опытом)	Владеет методологией построения эффективных алгоритмических решений для обработки больших данных в агробиоинформатике;	Не владеет:.	Владеет (на минимальном уровне, хорошо или отлично):	Итоговое тестирование, опрос, проверка выполненных работ, отчет		
			навыками работы с основными форматами NGS-данных и инструментами их первичной обработки и статистического анализа; опытом использования высокопроизводительных и облачных вычислительных ресурсов для запуска ресурсоёмких пайплайнов; приёмами оптимизации кода применительно к биологическим задачам; способностью самостоятельно проектировать вычислительный эксперимент, оценивать его воспроизводимость и проверять статистическую значимость результатов на больших выборках	– методологией построения эффективных алгоритмических решений для обработки больших данных в агробиоинформатике;	– навыками работы с основными форматами NGS-данных и инструментами их первичной обработки и статистического анализа;		– опытом использования высокопроизводительных и облачных вычислительных ресурсов для запуска ресурсоёмких пайплайнов;	– приёмами оптимизации кода применительно к биологическим задачам;

2. Структура учебной работы, содержание и трудоёмкость основных элементов дисциплины

2.1. Организационная структура, трудоёмкость и план изучения дисциплины

Дисциплина изучается в 1 семестре 1 курса.

Продолжительность семестра 19 недель.

Вид учебной работы	Трудоёмкость, час	
	1 семестр, 1 курс*	
	Очная форма обучения	
1. Контактная работа	60	
1.1. Аудиторные занятия, всего	36	
– лекции	18	
– практические занятия (включая семинары)	18	
– лабораторные работы	-	
1.2. Консультации (в соответствии с учебным планом)	24	
2. Внеаудиторная академическая работа	48	
2.1 Фиксированные виды внеаудиторных самостоятельных работ:	20	
Выполнение и сдача индивидуального задания в виде**		
– аналитического отчета	20	
2.2 Самостоятельное изучение тем/вопросов программы	8	
2.3 Самоподготовка к аудиторным занятиям	18	
2.4 Самоподготовка к участию и участие в контрольно-оценочных мероприятиях, проводимых в рамках текущего контроля освоения дисциплины <i>(за исключением учтённых в пп. 2.1 – 2.2):</i>	2	
3. Получение зачёта по итогам освоения дисциплины	+	
ОБЩАЯ трудоёмкость дисциплины:	Часы	108
	Зачетные единицы	3

Примечание:

* – **семестр** – для очной и очно-заочной формы обучения, **курс** – для заочной формы обучения;

** – КР/КП, реферата/эссе/презентации, контрольной работы (для обучающихся заочной формы обучения), расчетно-графической (расчетно-аналитической) работы и др.;

2.2. Укрупнённая содержательная структура учебной дисциплины и общая схема её реализации в учебном процессе

Номер и наименование раздела дисциплины. Укрупненные темы раздела		Общая	Трудоемкость раздела и ее распределение по видам учебной работы, час.							Формы текущего контроля успеваемости и промежуточной аттестации	№№ компетенций, на формирование которых ориентирован раздел	
			Контактная работа						ВАРС			
			Аудиторная работа				Консультации (в соответствии с учебным планом)					
			всего	лекции	занятия			всего	Фиксированные виды			
					практические (всех форм)	лабораторные						
1	1. Введение в алгоритмы и большие данные в агробиотехнологиях	8	4	2	2	-	2	2	-	Итоговое тестирование , опрос, проверка выполненных работ, отчет	ОПК- 3	
	2. Структуры данных для хранения и индексирования биологических последовательностей	8	4	2	2	-	2	2	-			
	3. Алгоритмы выравнивания последовательностей	8	4	2	2	-	2	2	-			
	4. Графовые алгоритмы в сборке геномов	8	4	2	2	-	2	2	-			
	5. Деревья и алгоритмы филогенетического анализа	12	4	2	2	-	4	4	-			
	6. Обработка потоковых данных и работа вне оперативной памяти	22	4	2	2	-	4	14	10			
	7. Параллельные вычисления и парадигма MapReduce	20	4	2	2	-	2	14	10			
	8. Алгоритмы поиска генетических маркеров и анализа вариативности	12	4	2	2	-	4	4	-			
	9. Интеграция и масштабирование алгоритмов машинного обучения для больших данных	10	4	2	2	-	2	4	-			
	Промежуточная аттестация	-	×	×	×	×	×	×	×	Зачет		
Итого по дисциплине		108		18	18		24	48	20			

3. Общие организационные требования к учебной работе обучающегося

3.1. Организация занятий и требования к учебной работе обучающегося

Организация занятий по дисциплине носит циклический характер. По трем разделам предусмотрена взаимоувязанная цепочка учебных работ: лекция – самостоятельная работа обучающихся (аудиторная и внеаудиторная). На занятиях студенческая группа получает задания и рекомендации.

Для своевременной помощи обучающимся при изучении дисциплины кафедрой организуются индивидуальные и групповые консультации, устанавливается время приема выполненных работ.

Учитывая статус дисциплины к её изучению предъявляются следующие организационные требования:

- обязательное посещение обучающимся всех видов аудиторных занятий;
- ведение конспекта в ходе лекционных занятий;
- качественная самостоятельная подготовка к практическим занятиям, активная работа на них;
- активная, ритмичная самостоятельная аудиторная и внеаудиторная работа обучающегося, своевременная сдача преподавателю отчетных документов по аудиторным и внеаудиторным видам работ;
- в случае наличия пропущенных обучающимся занятий, необходимо получить консультацию по подготовке и оформлению отдельных видов заданий.

Для успешного освоения дисциплины, обучающемуся предлагаются учебно-информационные источники в виде учебной, учебно-методической литературы по всем разделам.

4. Лекционные занятия

Для изучающих дисциплину читаются лекции в соответствии с планом, представленным в таблице 3.

Таблица 3 – Лекционный курс

№		Тема лекции. Основные вопросы темы	Трудоемкость по разделу, час.	Применяемые интерактивные формы обучения	
раздела	лекции				
1	1	Тема 1: Введение в алгоритмы и большие данные в агробиотехнологиях Роль больших данных в современном растениеводстве и животноводстве. Обзор типов данных (секвенирование геномов, транскриптомика, метагеномика почв, данные фенотипирования). Асимптотическая сложность алгоритмов (O-большое). Ограничения классических алгоритмов при работе с большими данными.	2	Лекция с ошибкой	
	2	Тема 2: Структуры данных для хранения и индексирования биологических последовательностей Строки, массивы, хеш-таблицы. Суффиксные деревья и суффиксные массивы. Преобразование Берроуза – Уилера (BWT) и индекс FM – основа современных алгоритмов поиска по геному.	2	Лекция визуализация – дискуссия	
	3	Тема 3: Алгоритмы выравнивания последовательностей Глобальное и локальное выравнивание (алгоритмы Нидлмана-Вунша и Смита-Ватермана). Эвристические алгоритмы для больших данных (Seed-and-extend). Принципы работы выравнивателей BWA, Bowtie2.	2	Лекция визуализация – дискуссия	
	4	Тема 4: Графовые алгоритмы в сборке геномов Графы и их представление в памяти. Сборка геномов на основе графов де Брёйна. Поиск эйлерова пути. Укладка графов (Graph layout) и устранение ошибок секвенирования.	2	Лекция визуализация – дискуссия	
	5	Тема 5: Деревья и алгоритмы филогенетического анализа Деревья (бинарные, мультиветвящиеся). Алгоритмы построения филогенетических деревьев (UPGMA, Neighbor-Joining). Применение филогенетики в животноводстве (анализ родословных, популяционная генетика пород) и растениеводстве (эволюция сортов).	2	Лекция визуализация – дискуссия	
	6	Тема 6: Обработка потоковых данных и работа вне оперативной памяти Парадигма потоковой обработки. Итераторы и генераторы в Python. Файлы с отображением в памяти (mmap). Обработка форматов VCF/BAM по частям (chunks) без полной загрузки в оперативную память	2	Лекция визуализация – дискуссия	
	7	Тема 7: Параллельные вычисления и парадигма MapReduce Параллелизм и конкуренция. Модель MapReduce. Применение MapReduce в биоинформатике (например, подсчет k-меров для всего генома пшеницы). Знакомство с фреймворком Apache Spark и его оболочкой на Python – PySpark.	2	Лекция визуализация – дискуссия	
	8	Тема 8: Алгоритмы поиска генетических маркеров и анализа вариативности Поиск паттернов в строках (алгоритм Бойера – Мура). Интервальные деревья (деревья интервалов) для работы с геномными аннотациями. Алгоритмы поиска однонуклеотидных полиморфизмов, инделов и структурных вариаций. Основы полногеномного поиска ассоциаций (Genome-Wide Association Study, GWAS) для поиска признаков (например, урожайности, молочной продуктивности).	2	Лекция визуализация – дискуссия	
	9	Тема 9: Интеграция и масштабирование алгоритмов машинного обучения для больших данных Разреженные матрицы (Sparse matrices) в геномике. Масштабирование алгоритмов кластеризации (K-Means, DBSCAN) и снижения размерности (PCA) для матриц SNP-маркеров (тысячи особей × миллионы полиморфизмов).	2	Лекция визуализация – дискуссия	
Общая трудоемкость лекционного курса			18	x	
Всего лекций по дисциплине:		час.	Из них в интерактивной форме:		час.
Очная форма обучения		18	Очная форма обучения		18

5. Практические занятия по дисциплине и подготовка к ним

Практические занятия по курсу проводятся в соответствии с планом, представленным в таблице 4.

Таблица 4 – Примерный тематический план практических занятий по разделам учебной дисциплины

№		Тема занятия / Примерные вопросы на обсуждение (для семинарских занятий)	Трудоемкость по разделу, час.	Используемые интерактивны е формы	Связь занятия с ВАРС*
раздела (модуля)	занятия				
1	1	Тема 1: Введение в алгоритмы и большие данные в агробиотехнологиях	2	-	ОСП
		Оценка сложности алгоритмов. Профилирование Python-кода (модули timeit, cProfile). Работа с памятью при загрузке больших геномных файлов.			
	2	Тема 2: Структуры данных для хранения и индексирования биологических последовательностей	2	Решение ситуационных задач Соревнование по бенчмаркингу	ОСП
		Реализация алгоритма BWT на Python. Создание словарей k-меров для генома растения с использованием эффективных структур данных.			
	3	Тема 3: Алгоритмы выравнивания последовательностей	2	-	ОСП
		Программная реализация динамического программирования для локального выравнивания. Оптимизация матриц с использованием библиотеки NumPy.			
	4	Тема 4: Графовые алгоритмы в сборке геномов	2	-	-
		Построение графа по набору ридов (например, бактерий или органелл растений) на Python. Поиск эйлера пути с использованием библиотеки NetworkX.			
	5	Тема 5: Деревья и алгоритмы филогенетического анализа	2	-	ОСП
		Реализация алгоритма UPGMA. Парсинг и визуализация деревьев в формате Newick с помощью библиотек Biopython и ETE3.			
	6	Тема 6: Обработка потоковых данных и работа вне оперативной памяти	2	Решение ситуационн ых задач Соревнование по бенчмаркингу	-
		Разработка скрипта для парсинга огромного VCF-файла (варианты сельскохозяйственных животных) с использованием генераторов. Подсчет частот аллелей без выгрузки всего файла в память.			
	7	Тема 7: Параллельные вычисления и парадигма MapReduce	2	Решение ситуационных задач Соревнование по бенчмаркингу	ОСП
		Использование библиотеки Dask для параллельной обработки и фильтрации большого набора геномных данных. Сравнение времени выполнения с базовым Pandas.			
	8	Тема 8: Алгоритмы поиска генетических маркеров и анализа вариативности	2	Решение ситуационных задач	-
		Использование библиотеки pybedtools для пересечения интервалов (поиск генов, попадающих в QTL-регионы количественных признаков у растений).			
	9	Тема 9: Интеграция и масштабирование алгоритмов машинного обучения для больших данных	2	-	ОСП
		Разработка пайплайна на Python (с использованием scikit-learn и scikit-allel): загрузка разреженной SNP-матрицы породы крупного рогатого скота, применение метода главных компонент для выявления популяционной структуры, оптимизация алгоритма по времени и памяти. Защита итоговых проектов.			
Всего практических занятий по дисциплине:		час.	Из них в интерактивной форме:		час.
Очная форма обучения		18	Очная форма обучения		8
* Условные обозначения: ОСП – предусмотрена обязательная самоподготовка к занятию; УЗ СРС – на занятии выдается задание на конкретную ВАРС; ПР СРС – занятие содержательно базируется на результатах выполнения обучающимся конкретной ВАРС.					

Подготовка обучающихся к практическим занятиям осуществляется с учетом общей структуры учебного процесса. На практических занятиях осуществляется входной и текущий аудиторный контроль в виде опроса, по основным понятиям дисциплины.

Для осуществления работы по подготовке к занятиям, необходимо ознакомиться с методическими указаниями по дисциплине, внимательно ознакомиться с литературой и электронными ресурсами.

6. Общие методические рекомендации по изучению отдельных разделов дисциплины

При изучении конкретного раздела дисциплины, из числа вынесенных на лекционные и практические занятия, обучающемуся следует учитывать изложенные ниже рекомендации. Обратите на них особое внимание при подготовке к аттестации.

Работа по теме прежде всего предполагает ее изучение по учебнику или пособию. Следует обратить внимание на то, что в любой теории, есть либо неубедительные, либо чересчур абстрактные, либо сомнительные положения. Поэтому необходимо вырабатывать самостоятельные суждения, дополняя их аргументацией, что и следует демонстрировать на семинарах. Для выработки самостоятельного суждения важным является умение работать с научной литературой. Поэтому работа по теме кроме ее изучения по учебнику, пособию предполагает также поиск по теме научных статей в научных журналах. Выбор статьи, относящейся к теме, лучше делать по последним в году номерам, где приводится перечень статей, опубликованных за год.

Самостоятельная подготовка предполагает использование ряда методов.

1. Конспектирование. Конспектирование позволяет выделить главное в изучаемом материале и выразить свое отношение к рассматриваемой автором проблеме.

Техника записей в конспекте индивидуальна, но есть ряд правил, которые могут принести пользу его составителю: начиная конспект, следует записать автора изучаемого произведения, его название, источник, где оно опубликовано, год издания. Порядок конспектирования:

- а) внимательное чтение текста;
- б) поиск в тексте ответов на поставленные в изучаемой теме вопросы;
- в) краткое, но четкое и понятное изложение текста;
- г) выделение в записи наиболее значимых мест;
- д) запись на полях возникающих вопросов, понятий, категорий и своих мыслей.

2. Записи в форме тезисов, планов, аннотаций, формулировок определений. Все перечисленные формы помогают быстрой ориентации в подготовленном материале, подборе аргументов в пользу или против какого-либо утверждения.

3. Словарь понятий и категорий. Составление словаря помогает быстрее осваивать новые понятия и категории, увереннее ими оперировать. Подобный словарь следует вести четко, разборчиво, чтобы удобно было им пользоваться. Из приведенного в УМК глоссария нужно к каждому семинару выбирать понятия, относящиеся к изучаемой теме, объединять их логической схемой в соответствии с вопросами семинарского занятия.

Внеаудиторной работой, является проектно-ориентированной работой на тему «Оптимизация алгоритма обработки больших биологических данных».

Студенту предлагается «проблемная ситуация»: есть базовый алгоритм решения задачи, написанный на Python, в котором допущена ошибка или он работает очень долго. Задача обучающегося заключается в проведении анализа алгоритмической сложности, подборе правильной структуры данных или корректировка представленного кода.

Тема 1. Введение в алгоритмы и большие данные в агробиотехнологиях

Краткое содержание

Роль больших данных в современном растениеводстве и животноводстве. Обзор типов данных (секвенирование геномов, транскриптомика, фенотипирование). Понятие алгоритмической сложности: временная и пространственная. Асимптотические обозначения. Ограничения классических алгоритмов при работе с Big Data. Инструменты профилирования кода в Python.

Вопросы для самоконтроля по разделу

1. Какие объемы данных генерируются при секвенировании генома полиплоидной пшеницы?
2. В чем разница между временной и пространственной сложностью алгоритма?
3. Объясните разницу между сложностями $O(N)$, $O(N \log N)$ и $O(N^2)$ на примере обработки списка генов.
4. Почему алгоритм с сложностью $O(2^N)$ непригоден для анализа ДНК?
5. Какие инструменты Python используются для измерения времени выполнения и потребления памяти скрипта?

Процедура оценивания:

Выполнение практического задания по анализу сложности предоставленного фрагмента кода и применение профилировщиков ('timeit', 'memory_profiler') для поиска узких мест.

Шкала и критерии оценивания:

Зачтено: студент верно определил асимптотическую сложность фрагментов кода, корректно применил профилировщики, предоставил график потребления ресурсов и сделал правильный вывод о применимости алгоритма для больших данных.

Не зачтено: допущены ошибки в определении сложности циклов/рекурсий, не удалось запустить инструменты профилирования, отсутствует анализ результатов.

Тема 2. Структуры данных для хранения и индексирования биологических последовательностей

Краткое содержание

Специфика хранения строк (ДНК/белков) в памяти. Хеш-таблицы и их применение для подсчета k-меров. Суффиксные массивы. Преобразование Барроуза-Уилера (BWT) и индекс FM — математическая основа современных индексов генома.

Вопросы для самоконтроля по разделу

1. Почему для хранения генома в оперативной памяти не всегда оптимально использовать обычный тип `str` в Python?
2. В чем преимущество хеш-таблиц перед списками при поиске специфических SNP-маркеров?
3. Опишите алгоритм построения BWT для короткой строки.
4. Каким образом BWT позволяет сжимать геномные данные?
5. За счет чего индекс FM обеспечивает быстрый поиск подстроки в геноме?

Процедура оценивания:

Написание программы на Python для реализации алгоритма BWT и создания словаря k-меров с оценкой потребляемой памяти.

Шкала и критерии оценивания:

Зачтено: код корректно реализует алгоритм BWT (или эффективный подсчет k-меров с использованием подходящих структур данных), работает без ошибок на тестовых строках, студент может объяснить логику работы алгоритма.

Не зачтено: алгоритм реализован неверно (нарушена логика сортировки/хеширования), код падает с ошибками, студент не может объяснить принцип работы выбранной структуры данных.

Тема 3. Алгоритмы выравнивания последовательностей

Краткое содержание

Глобальное выравнивание (алгоритм Нидлмана-Вунша) и локальное выравнивание (алгоритм Смита-Ватермана). Основы динамического программирования. Матрицы штрафов (BLOSUM, PAM). Эвристические алгоритмы для больших данных (Seed-and-extend). Принципы работы выравнивателей BWA и Bowtie2.

Вопросы для самоконтроля по разделу

1. В чем принципиальная разница между глобальным и локальным выравниванием?
2. Как заполняется матрица динамического программирования при поиске сходства двух генов?
3. Что такое штраф за гэп (gap penalty) и почему он необходим при сравнении геномов разных видов?
4. Почему алгоритм Смита-Ватермана нельзя напрямую применить для выравнивания ридов длиной 150 п.н. на геном человека?
5. В чем суть эвристики Seed-and-extend?

Процедура оценивания:

Самостоятельная программная реализация матрицы динамического программирования (локального выравнивания) с использованием библиотеки `NumPy` для оптимизации вычислений.

Шкала и критерии оценивания:

Зачтено: матрица DP построена верно, реализован трассировочный шаг (восстановление выравнивания), применена векторизация через `NumPy` (отсутствие медленных вложенных циклов Python), получен правильный результат на тестовом примере.

Не зачтено: нарушена рекуррентная формула DP, трассировка не реализована или реализована с ошибками, код работает катастрофически долго из-за неоптимизированных циклов.

Тема 4. Графовые алгоритмы в сборке геномов

Краткое содержание

Представление графов в памяти (матрица смежности, список ребер). Оверлап-графы. Графы де Брёйна: построение из набора ридов. Поиск эйлерова пути как основа современной сборки геномов (De novo assembly). Укладка графов и устранение пузырей (ошибок секвенирования).

Вопросы для самоконтроля по разделу

1. Почему для сборки генома из миллионов коротких ридов не используют попарное сравнение (BLAST)?
2. Как превратить набор ридов в граф де Брёйна? Что является вершинами, а что — ребрами?
3. Сформулируйте условия существования эйлерова пути в графе.
4. Чем эйлеров путь принципиально лучше гамильтонова пути для задачи сборки ДНК?
5. Что такое «пузырь» в графе де Брёйна и как он возникает биологически?

Процедура оценивания:

Построение графа де Брёйна на Python (с использованием `NetworkX`) по заданному набору синтетических ридов и поиск эйлерова пути.

Шкала и критерии оценивания:

Зачтено: граф построен корректно (k-1 меры как вершины), успешно найден эйлеров путь, восстановлена исходная последовательность. Студент понимает, как размер графа зависит от размера k.

Не зачтено: путаница между вершинами и ребрами при построении графа де Брёйна, не найден путь из-за игнорирования баланса степеней вершин, код не работает.

Тема 5. Деревья и алгоритмы филогенетического анализа

Краткое содержание

Деревья в информатике (бинарные деревья поиска, деревья с множественным ветвлением). Алгоритмы построения филогенетических деревьев: UPGMA и Neighbor-Joining. Применение филогенетики в животноводстве (анализ родословных, популяционная генетика) и растениеводстве (эволюция сортов). Форматы хранения деревьев (Newick).

Вопросы для самоконтроля по разделу

1. Чем кластерное дерево (дендограмма) отличается от графового дерева?
2. Опишите пошаговый алгоритм UPGMA. В каком случае он дает неверные результаты?
3. В чем преимущество метода Neighbor-Joining над UPGMA?
4. Как distance matrix используется на каждом шаге алгоритма кластеризации?
5. Какова роль филогенетического анализа при сертификации новых сортов растений?

Процедура оценивания:

Реализация алгоритма UPGMA на Python для заданной матрицы дистанций и визуализация результата с помощью библиотеки `Biopython` или `ETE3`.

Шкала и критерии оценивания:

Зачтено: матрица построена алгоритмически верно, дерево построено, выведено в формате Newick, визуализация корректна. Студент может интерпретировать ветви дерева.

Не зачтено: допущены ошибки в поиске минимального элемента матрицы, неверный пересчет дистанций после объединения кластеров, неумение работать с библиотеками вывода деревьев.

Тема 6. Обработка потоковых данных и работа вне оперативной памяти

Краткое содержание

Проблема `MemoryError` при работе с геномными файлами (VCF, BAM, FASTQ). Парадигма Stream Processing. Итераторы и генераторы в Python (`yield`). Технология Memory-mapped files (`mmap`). Обработка файлов по кускам (chunking) без их полной загрузки в RAM.

Вопросы для самоконтроля по разделу

1. Почему операция `pandas.read\_csv()` для большого VCF-файла приводит к краху программы?
2. В чем физическая разница между списком (list) и генератором (generator) в Python с точки зрения выделения памяти?
3. Как работает функция `yield`?
4. Что такое `mmap` и за счет чего она позволяет работать с файлами размером больше оперативной памяти?
5. Запишите псевдокод потокового подсчета строк в файле.

Процедура оценивания:

Рефакторинг (оптимизация) предоставленного «плохого» скрипта чтения VCF-файла: перевод его на рельсы генераторов для снижения потребления RAM в 100+ раз.

Шкала и критерии оценивания:

Зачтено: выполнен успешный рефакторинг, память при выполнении скрипта не растет пропорционально объему файла (остается константной $O(1)$), студент демонстрирует замеры памяти до и после.

Не зачтено: попытка решить проблему просто увеличением лимитов системы или использованием неэффективных структур, непонимание механики `yield`, скрипт по-прежнему потребляет много памяти.

Тема 7. Параллельные вычисления и парадигма MapReduce

Краткое содержание

Параллелизм (многопоточность vs мультипроцессинг в Python из-за GIL). Масштабирование горизонтальное и вертикальное. Парадигма MapReduce. Применение MapReduce для биоинформатики (пример: подсчет k-меров всего генома). Введение в фреймворк Apache Spark и его Python-обертку PySpark. Библиотека Dask для локального параллелизма.

Вопросы для самоконтроля по разделу

1. Почему в Python потоки (`threading`) не ускоряют вычислительные алгоритмы?
2. В чем суть парадигмы MapReduce? Приведите пример применительно к подсчету частоты нуклеотидов.
3. Что такое "Shuffle" этап в MapReduce?
4. Чем Dask DataFrame отличается от обычного Pandas DataFrame при работе на одном компьютере?
5. Как рабочие узлы (workers) обмениваются данными в кластере Spark?

Процедура оценивания:

Написание скрипта с использованием библиотеки `Dask` или `multiprocessing` для параллельной обработки множества геномных файлов и сравнение времени выполнения с однопоточным вариантом.

Шкала и критерии оценивания:

Зачтено: код успешно распараллелен, продемонстрирован реальный прирост производительности (близкий к кратному числу ядер), студент понимает разницу между Map и Reduce этапами.

Не зачтено: использование `threading` для CPU-bound задач (что не дает ускорения), ошибки синхронизации процессов, код падает или работает медленнее однопоточного варианта.

Тема 8. Алгоритмы поиска генетических маркеров и анализа вариативности

Краткое содержание

Эффективный поиск паттернов в строках (алгоритм Бойера-Мура). Интервальные деревья (Interval Trees) для перекрытия геномных аннотаций. Алгоритмы извлечения SNP и инделей. Поиск пересечений генетических маркеров с QTL-регионами (локусы количественных признаков) у растений и животных. Основы формата BED и утилиты `bedtools`.

Вопросы для самоконтроля по разделу

1. За счет чего алгоритм Бойера-Мура позволяет пропускать символы при поиске подстроки?
2. Почему для задачи «найти все гены, попадающие в интервал хромосомы от 1Мб до 2Мб» не подходит линейный поиск по списку?
3. Как устроено интервальное дерево? Какова сложность поиска в нем?
4. Что такое QTL-анализ и как он связан с селекцией сельскохозяйственных животных?
5. Как алгоритмически происходит пересечение двух наборов геномных интервалов?

Процедура оценивания:

Применение библиотеки `pybedtools` (или аналогичной на Python) для быстрого поиска пересечений между списком SNPs и областями генома, отвечающими за стрессоустойчивость у растений.

Шкала и критерии оценивания:

Зачтено: задача пересечения решена с использованием специализированных алгоритмов интервалов (а не медленных вложенных циклов), код работает быстро на больших выборках, студент понимает биологический смысл найденных пересечений.

Не зачтено: реализован "наивный" алгоритм попарного сравнения с сложностью $O(N^2)$, который зависит от тестовых данных, непонимание принципов работы интервальных деревьев.

Тема 9. Интеграция и масштабирование алгоритмов машинного обучения для больших данных

Краткое содержание

Разреженные матрицы (Sparse matrices) в геномике (матрицы SNP: особь x полиморфизм). Масштабирование алгоритмов кластеризации (K-Means) и снижения размерности (PCA) для матриц размером 10 000 x 1 000 000. Использование `scikit-allel` и интеграция с `Dask-ml`. Применение машинного обучения для выявления популяционной структуры стада или сортов.

Вопросы для самоконтроля по разделу

1. Почему матрицу SNP-маркеров нельзя хранить в виде обычного массива `NumPy` (dense array)?
2. Какой формат хранения разреженных матриц (CSR, CSC) оптимален для алгоритма PCA и почему?
3. В чем заключается проблема масштабирования алгоритма K-Means при увеличении числа особей в датасете?
4. Как `Dask-ml` позволяет запустить PCA на данных, не помещающихся в RAM?
5. Как результаты PCA помогают селекционеру в управлении животноводческим стадом?

Процедура оценивания:

Разработка пайплайна: загрузка разреженной SNP-матрицы -> выполнение масштабируемого PCA (через `scikit-allel`/`Dask`) -> визуализация кластеров. Оценка потребления RAM.

Шкала и критерии оценивания:

Зачтено: данные загружены в разреженном формате, алгоритм PCA успешно выполнен без переполнения памяти, получена визуализация (scatter plot) кластеров, студент сделал вывод о популяционной структуре.

Не зачтено: попытка преобразовать разреженную матрицу в плотную (что привело к MemoryError), неверное использование библиотек, отсутствие биологической интерпретации графиков кластеризации.

7. Общие методические рекомендации по оформлению и выполнению отдельных видов ВАРС

7.1. Рекомендации по написанию отчета

Учебные цели, на достижение которых ориентировано выполнение отчета: формирование навыков и алгоритмического мышления, направленного на преодоление физических ограничений вычислительных систем (недостаток оперативной памяти, процессорного времени) при решении прикладных задач геномики, селекции и биотехнологий сельскохозяйственных культур..

Учебные задачи, которые должны быть решены обучающимся в рамках выполнения реферата:

- абстрактное мышление и формализация;
- формирование и отработка навыков алгоритмической критики и доказательного обоснования принятых решений при выполнении поставленной задачи;
- архитектурное проектирование, самостоятельного построения структуры работы, постановки задач, раскрытие основных вопросов, умение сформулировать логические выводы и предложения;
- преодолению разрыва между информационными технологиями и предметной областью.

ПРИМЕРНАЯ ТЕМАТИКА

Отчет выполняется по теме: «**Оптимизация алгоритма обработки больших биологических данных**» по вариантам, выдаваемых преподавателем

Этапы работы над отчетом

Студенту предлагается «проблемная ситуация»: есть базовый алгоритм решения задачи, написанный на Python, который заведомо некорректно работает. Задача обучающегося – проанализировать алгоритмическую сложность, подобрать правильную структуру данных или парадигму (изученную на лекциях) и переписать код так, чтобы он справлялся с большими данными.

Исходные данные

Преподаватель выдает студенту:

- индивидуальный вариант (например, «Поиск SNP-маркеров мастита у крупного рогатого скота»);
- базовый Python-скрипт, который некорректно решает задачу на небольших массивах данных;
- набор данных: усеченный датасет; ссылка на полный датасет Big Dat.

Порядок выполнения

Этап 1. Формализация и анализ (аналитическая часть)

1. Опишите входные данные и желаемый результат в математическом виде.
2. Проанализируйте предложенный «базовый» алгоритм.

Этап 2. Разработка нового алгоритма (проектная часть)

1. Выбрать подходящую структуру данных или парадигму обработки из изученных в курсе (потокковая обработка, хеш-таблицы с открытой адресацией, графы де Брёйна, интервальные деревья, разреженные матрицы, парадигма MapReduce и т. д.).
2. Разработать формальный оптимизированный алгоритм.
3. Обосновать выбор структуры данных. Оценить новую сложность по времени и использованию памяти. Доказать, что новая сложность позволяет обрабатывать большие данные в заданных ограничениях по ресурсам.

Этап 3. Программная реализация (практическая часть)

1. Написать оптимизированный код на языке Python (допускается использование стандартных библиотек для работы с памятью и параллелизма: sys, mmap, collections, dask, pyspark, scipy.sparse, pybedtools и др.).
2. Код должен быть модульным, сопровождаться документацией и типизацией данных.
3. Провести профилирование (cProfile, memory_profiler) старого и нового кода на небольшом наборе данных.

Этап 4. Тестирование на больших данных

1. Запустить оптимизированный алгоритм на полном наборе данных Big Data. Зафиксировать потребление оперативной памяти и время выполнения.

3. Сравнить показатели старого и нового подходов (для старого подхода провести теоретический расчет или экстраполяцию времени сбой).

Форма отчета

Отчет представляется в виде документа и содержит следующие разделы:

1. Титульный лист (ФИО, группа, вариант).
2. Постановка задачи (биологический контекст и формальное математическое описание).
3. Анализ исходного алгоритма (блок-схема или псевдокод наивного решения, расчет сложности, обоснование причины неспособности обработать большие данные).
4. Разработка оптимизированного алгоритма (обоснование выбора структур данных, псевдокод или блок-схема нового решения, строгий математический расчет новой сложности).
5. Результаты тестирования (бенчмарки):
 - Таблица с результатами замеров времени и объема памяти.
 - Графики масштабирования (зависимость времени/объема памяти от объема входных данных).
 - Биологическая интерпретация (краткое описание биологического смысла полученных выходных данных для агробиотехнологий).
6. Ссылка на репозиторий с исходным кодом, файлом requirements.txt и инструкцией по запуску (README.md).

Варианты заданий (ВАРС)

Каждый вариант строго привязан к растениеводству или животноводству и определенному модулю дисциплины.

Вариант 1. Геномика полиплоидных растений (тема 2 и 6)

Биологическая задача: Подсчет уникальных k-меров (например, 21-нуклеотидных последовательностей) в геноме современной сортовой пшеницы (*Triticum aestivum*), которая имеет огромный размер (~15-17 Гб) и является гексаплоидом.

Проблема: Дан базовый код, который включает все k-меры в стандартный словарь Python (dict), что приводит к переполнению оперативной памяти.

Задание студента: Переписать алгоритм с использованием структур, учитывающих хеширование с открытой адресацией, или применить парадигму потоковой обработки (Stream processing) и сортировки внешней памятью (External sorting), чтобы уложиться в заданные лимиты RAM (например, 4 Гб).

Вариант 2. Поиск маркеров устойчивости к болезням у животных (тема 6 и 8)

Биологическая задача: Поиск специфических мутаций (однонуклеотидных полиморфизмов), связанных с устойчивостью к маститу у крупного рогатого скота, в большом VCF-файле (вызов вариантов), содержащем генотипы 10 000 особей.

Проблема: Исходный код использует библиотеку pandas для загрузки всего VCF в DataFrame, что приводит к сбою системы.

Задание для студента: Реализовать алгоритм построчного парсинга VCF с использованием генераторов Python. Применить структуру данных «Дерево интервалов» (Interval Tree) для быстрого поиска пересечений найденных SNP с annotated QTL-регионами (количественными признаками) в геноме коровы.

Вариант 3. Анализ популяционной структуры стада (модули 7 и 9)

Биологическая задача: Кластеризация породы свиней на основе матрицы SNP-маркеров (50 000 свиней × 100 000 SNP) для выявления скрытых родственных линий.

Проблема: Стандартный алгоритм K-средних или расчет матрицы расстояний с помощью библиотеки требуют создания плотной матрицы размером 50 000 × 100 000, что невозможно сделать классическими методами.

Задание для студента: Перевести данные в формат разреженных матриц. Разработать распределенный алгоритм с использованием парадигмы, распараллелив вычисление расстояний между особями.

Титульный лист заполняется по единой форме (приложение 1).

Процедура оценивания

При аттестации обучающегося по итогам его работы над отчетом, руководителем используются критерии оценки качества **процесса подготовки отчета**, критерии оценки **содержания отчета**, критерии оценки **оформления отчета**, критерии оценки **участия обучающегося в контрольно-оценочном мероприятии**.

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ

Зачтено: корректно определена временная и пространственная сложность обоих алгоритмов. Студент свободно оперирует терминами, связанными со структурами данных и парадигмами обработки. Задача формализована математически. Предложен корректный алгоритм оптимизации. Код написан чисто, запускается и выдает верный биологический результат. Алгоритм успешно обработал полный набор данных Big Data без превышения заданных лимитов оперативной памяти.

Не зачтено: не корректно определена временная и пространственная сложность обоих алгоритмов. Студент не оперирует терминами, связанными со структурами данных и парадигмами обработки. Предложен не корректный алгоритм оптимизации, код не запускается и выдает неверный биологический результат.

Оценка по отчету расписывается преподавателем в оценочном листе (приложение 2)

7.2. Рекомендации по самостоятельному изучению тем

ВОПРОСЫ

для самостоятельного изучения темы

«Задачи биоинформатики»

1. Классификация основных вычислительных задач биоинформатики: сборка геномов de novo и выравнивание ридов (с учетом специфики работы с полиплоидными геномами растений и высокогетерозиготными геномами сельскохозяйственных животных).

2. Изучение задач структурной и функциональной аннотации геномов: алгоритмическое предсказание генов, поиск регуляторных элементов promoters/enhancers, определение функциональности белков (с акцентом на поиск генов, ответственных за признаки хозяйственно-полезных свойств).

3. Анализ генетической изменчивости как источник больших данных: задачи выявления полиморфизмов (SNP, инделей, структурных вариаций), популяционный анализ (определение генетической структуры стада или коллекции сортов) и ассоциативные исследования (GWAS).

ВОПРОСЫ

для самостоятельного изучения темы

«Алгоритмы машинного обучения для поиска структур в генетических данных. Нейронные сети»

1. Специфика применения методов машинного обучения (классификация, кластеризация, регрессия) в агрогеномике по сравнению с классическими задачами машинного зрения или обработки естественного языка. Приведите примеры предсказания фенотипических признаков по генотипу.

2. Расскажите архитектуру и принцип работы сверточных нейронных сетей (CNN) применительно к анализу одномерных последовательностей ДНК/РНК? Как CNN решает задачу поиска мотивов (motif finding), таких как сайты связывания транскрипционных факторов?

3. Какие алгоритмические и вычислительные проблемы возникают при обучении нейронных сетей на генетических матрицах размерностью «тысячи особей × миллионы полиморфизмов», и какие методы оптимизации применяются для их решения?

Общий алгоритм самостоятельного изучения темы

1) Ознакомиться с рекомендованной учебной литературой и электронными ресурсами по теме (ориентируясь на вопросы для самоконтроля).
2) На этой основе составить развёрнутый план изложения темы
3) Провести самоконтроль освоения темы по вопросам, выданным преподавателем
4) Подготовиться к предусмотренному контрольно-оценочному мероприятию по результатам самостоятельного изучения темы
4) Принять участие в указанном мероприятии

ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ самостоятельного изучения темы

Оценка «зачтено» выставляется, если студент на основе самостоятельного изученного материала, смог всесторонне раскрыть теоретическое содержание темы.

Оценка «не зачтено» выставляется, если студент на основе самостоятельного изученного материала, не смог всесторонне раскрыть теоретическое содержание темы.

8. Входной контроль и текущий (внутрисеместровый) контроль хода и результатов учебной работы

8.1. Вопросы для входного контроля

Не предусмотрен.

8.2. Текущий контроль успеваемости

В течение семестра, проводится текущий контроль успеваемости по дисциплине, к которому обучающийся должен быть подготовлен.

Отсутствие пропусков аудиторных занятий, активная работа на практических занятиях, общее выполнение графика учебной работы являются основанием для получения положительной оценки по текущему контролю.

ВОПРОСЫ и ЗАДАЧИ для самоподготовки к семинарским занятиям

В процессе подготовки к семинарскому занятию обучающийся изучает представленные ниже вопросы по темам. На занятии обучающийся демонстрирует свои знания по изученным вопросам в форме устного ответа.

Тема 1. Введение в алгоритмы и большие данные в агrobiотехнологиях

1. Какие объемы данных генерируются при секвенировании генома полиплоидной пшеницы?
2. В чем разница между временной и пространственной сложностью алгоритма?
3. Объясните разницу между сложностями $O(N)$, $O(N \log N)$ и $O(N^2)$ на примере обработки списка генов.
4. Почему алгоритм с сложностью $O(2^N)$ непригоден для анализа ДНК?
5. Какие инструменты Python используются для измерения времени выполнения и потребления памяти скрипта?

Тема 2. Структуры данных для хранения и индексирования биологических последовательностей

1. Почему для хранения генома в оперативной памяти не всегда оптимально использовать обычный тип `str` в Python?
2. В чем преимущество хеш-таблиц перед списками при поиске специфических SNP-маркеров?
3. Опишите алгоритм построения BWT для короткой строки.
4. Каким образом BWT позволяет сжимать геномные данные?
5. За счет чего индекс FM обеспечивает быстрый поиск подстроки в геноме?

Тема 3. Алгоритмы выравнивания последовательностей

1. В чем принципиальная разница между глобальным и локальным выравниванием?
2. Как заполняется матрица динамического программирования при поиске сходства двух генов?
3. Что такое штраф за гэп (gap penalty) и почему он необходим при сравнении геномов разных видов?
4. Почему алгоритм Смита-Ватермана нельзя напрямую применить для выравнивания ридов длиной 150 п.н. на геном человека?
5. В чем суть эвристики Seed-and-extend?

Тема 4. Графовые алгоритмы в сборке геномов

1. Почему для сборки генома из миллионов коротких ридов не используют попарное сравнение (BLAST)?
2. Как превратить набор ридов в граф де Брёйна? Что является вершинами, а что — ребрами?
3. Сформулируйте условия существования эйлера пути в графе.
4. Чем эйлеров путь принципиально лучше гамильтонова пути для задачи сборки ДНК?
5. Что такое «пузырь» в графе де Брёйна и как он возникает биологически?

Тема 5. Деревья и алгоритмы филогенетического анализа

1. Чем кластерное дерево (дендограмма) отличается от графового дерева?
2. Опишите пошаговый алгоритм UPGMA. В каком случае он дает неверные результаты?
3. В чем преимущество метода Neighbor-Joining над UPGMA?
4. Как distance matrix используется на каждом шаге алгоритма кластеризации?
5. Какова роль филогенетического анализа при сертификации новых сортов растений?

Тема 6. Обработка потоковых данных и работа вне оперативной памяти

1. Почему операция `pandas.read_csv()` для большого VCF-файла приводит к краху программы?
2. В чем физическая разница между списком (list) и генератором (generator) в Python с точки зрения выделения памяти?
3. Как работает функция `yield`?
4. Что такое `mmap` и за счет чего она позволяет работать с файлами размером больше оперативной памяти?
5. Запишите псевдокод потокового подсчета строк в файле.

Тема 7. Параллельные вычисления и парадигма MapReduce

1. Почему в Python потоки (`threading`) не ускоряют вычислительные алгоритмы?
2. В чем суть парадигмы MapReduce? Приведите пример применительно к подсчету частоты нуклеотидов.
3. Что такое "Shuffle" этап в MapReduce?
4. Чем Dask DataFrame отличается от обычного Pandas DataFrame при работе на одном компьютере?
5. Как рабочие узлы (workers) обмениваются данными в кластере Spark?

Тема 8. Алгоритмы поиска генетических маркеров и анализа вариативности

1. За счет чего алгоритм Бойера-Мура позволяет пропускать символы при поиске подстроки?
2. Почему для задачи «найти все гены, попадающие в интервал хромосомы от 1Мб до 2Мб» не подходит линейный поиск по списку?
3. Как устроено интервальное дерево? Какова сложность поиска в нем?
4. Что такое QTL-анализ и как он связан с селекцией сельскохозяйственных животных?
5. Как алгоритмически происходит пересечение двух наборов геномных интервалов?

Тема 9. Интеграция и масштабирование алгоритмов машинного обучения для больших данных

1. Почему матрицу SNP-маркеров нельзя хранить в виде обычного массива `NumPy` (dense array)?
2. Какой формат хранения разреженных матриц (CSR, CSC) оптимален для алгоритма PCA и почему?
3. В чем заключается проблема масштабирования алгоритма K-Means при увеличении числа особей в датасете?
4. Как `Dask-ml` позволяет запустить PCA на данных, не помещающихся в RAM?
5. Как результаты PCA помогают селекционеру в управлении животноводческим стадом?

Общий алгоритм самоподготовки

1. Рассмотрение вопросов семинара.
2. Изучение литературы по вопросам семинара.
3. Подготовка ответов на вопросы.

Шкала и критерии оценивания самоподготовки по темам семинарских занятий

Оценка «зачтено» выставляется обучающемуся, если все вопросы темы раскрыты, во время дискуссии высказывается собственная точка зрения на обсуждаемую проблему, демонстрируется способность аргументировать доказываемые положения и выводы.

Оценка «не зачтено» выставляется, если обучающийся не способен доказать и аргументировать собственную точку зрения по изученной теме, не способен сослаться на мнения ведущих специалистов по обсуждаемой проблеме.

9. Промежуточная (семестровая) аттестация по курсу

Нормативная база проведения промежуточной аттестации обучающихся по результатам изучения дисциплины:	
Действующее «Положение о текущем контроле успеваемости, промежуточной аттестации обучающихся по программам высшего образования (бакалавриат, специалитет, магистратура) и среднего профессионального образования в ФГБОУ ВО Омский ГАУ»	
Основные характеристики промежуточной аттестации обучающихся по итогам изучения дисциплины	
Цель промежуточной аттестации –	установление уровня достижения каждым обучающимся целей и задач обучения по данной дисциплине, изложенным в п.2.2 настоящей программы
Форма промежуточной аттестации –	Зачет
Место процедуры получения зачёта в графике учебного процесса	1) участие обучающегося в процедуре получения зачёта осуществляется за счёт учебного времени (трудоемкости), отведённого на изучение дисциплины
	2) процедура проводится в рамках ВАРО, на последней неделе семестра
Основные условия получения обучающимся зачёта:	1) обучающийся выполнил все виды учебной работы (включая самостоятельную) и отчитался об их выполнении в сроки, установленные графиком учебного процесса по дисциплине; 2) прошёл заключительное тестирование; 3) подготовил полнокомплектное учебное портфолио
Процедура получения зачёта – Методические материалы, определяющие процедуры оценивания знаний, умений, навыков:	Представлены в Фонде оценочных средств по данной учебной дисциплине (см. – Приложение 9)

9.2 Процедура проведения зачета

Зачет выставляется по итогам выполнения практических заданий, ответов на семинарских заданиях, предоставлении отчета по ВАРС, прохождении опроса по темам, вынесенным на самостоятельное изучение. По итогам изучения дисциплины обучающиеся проходят итоговое тестирование

9.3. Заключительное тестирование по итогам изучения дисциплины

По итогам изучения дисциплины, обучающиеся проходят заключительное тестирование. Тестирование является формой контроля, направленной на проверку владения терминологическим аппаратом и конкретными знаниями в области фундаментальных и прикладных дисциплин.

9.3.1 Подготовка к заключительному тестированию по итогам изучения дисциплины

Тестирование осуществляется по всем темам и разделам дисциплины, включая темы, выносимые на самостоятельное изучение. Процедура тестирования ограничена во времени и предполагает максимальное сосредоточение обучающегося на выполнении теста, содержащего несколько тестовых заданий.

Тестирование проводится в электронной / письменной форме (на бумажном носителе). Тест включает в себя 30 вопросов. Время, отводимое на выполнение теста - 30 минут. В каждый вариант теста включаются вопросы разных типов (одиночный и множественный выбор, открытые (ввод ответа с клавиатуры), на упорядочение, соответствие и др.). На тестирование выносятся вопросы из каждого раздела дисциплины.

Бланк теста

Образец

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Омский государственный аграрный университет имени П.А. Столыпина»

Тестирование по итогам освоения дисциплины «Алгоритмы и структуры больших данных» Для обучающихся направления подготовки 19.04.01 Биотехнология

ФИО _____ группа _____

Дата _____

Уважаемые обучающиеся!

Прежде чем приступить к выполнению заданий внимательно ознакомьтесь с инструкцией:

1. Отвечая на вопрос с выбором правильного ответа, правильный, на ваш взгляд, ответ (ответы) обведите в кружок.
2. В заданиях открытой формы впишите ответ в пропуск.
3. В заданиях на соответствие заполните таблицу.
4. В заданиях на правильную последовательность впишите порядковый номер в квадрат.
4. Время на выполнение теста – 30 минут
5. За каждый верный ответ Вы получаете 1 балл, за неверный – 0 баллов. Максимальное количество полученных баллов 30.

Желаем удачи!

Вариант № 1

1. Какой формат файла является стандартным для хранения нуклеотидных последовательностей с показателями качества каждого основания?

FASTA
FASTQ
SAM
VCF

2. Какая структура данных лежит в основе алгоритма сборки генома «граф де Брёйна»?

Двоичное дерево поиска
Хеш-таблица k-меров и ориентированный граф
Суффиксный массив
Список смежности белковых доменов

И т.д.

9.3.1 ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ

ответов на тестовые вопросы тестирования по итогам освоения дисциплины

Оценка «отлично» выставляется обучающемуся, если получено более 81% правильных ответов.

Оценка «хорошо» - получено от 71 до 80% правильных ответов.

Оценка «удовлетворительно» - получено от 61 до 70% правильных ответов.

Оценка «неудовлетворительно» - получено менее 61% правильных ответов.

Выставление оценки по итогам промежуточного контроля осуществляется с учетом описания показателей, критериев и шкал оценивания компетенций по дисциплине, представленных в таблице 1.2

10. Информационное и методическое обеспечение учебного процесса по дисциплине

В соответствии с действующими государственными требованиями для реализации учебного процесса по дисциплине обеспечивающей кафедрой разрабатывается и постоянно совершенствуется учебно-методический комплекс (УМКД), соответствующий данной рабочей программе и прилагаемый к ней. При разработке УМКД кафедра руководствуется установленными университетом требованиями к его структуре, содержанию и оформлению. В состав УМКД входят перечисленные ниже и другие источники учебной и учебно-методической информации, средства наглядности.

Электронная версия актуального УМКД, адаптированная для обучающихся, выставляется в электронной информационно-образовательной среде университета.

ПЕРЕЧЕНЬ литературы, рекомендуемой для изучения дисциплины Б1.О.09 Алгоритмы и структуры больших данных	
Автор, наименование, выходные данные	Доступ
Баланов, А. Н. Искусственный интеллект. Понимание, применение и перспективы : учебник для вузов / А. Н. Баланов. – 3-е изд., стер. – Санкт-Петербург : Лань, 2026. – 312 с. – ISBN 978-5-507-55902-2. – Текст : электронный // Лань : электронно-библиотечная система. – URL : https://e.lanbook.com/book/512029 . – Режим доступа : для авториз. пользователей	https://e.lanbook.com
Бождай, А.С. МАШИННОЕ ПРЕДСТАВЛЕНИЕ БОЛЬШИХ ГРАФОВЫХ МОДЕЛЕЙ / А. С. Бождай, С. В. Прошкин // Вестник Пензенского государственного университета. – 2021. – № 3. – С. 101-106. – ISSN 2410-2083. – Текст : электронный // Лань : электронно-библиотечная система. – URL : https://e.lanbook.com/journal/issue/315149 . – Режим доступа : для авториз. пользователей	https://e.lanbook.com
Высокогорский, В. Е. Биохимические основы биоинженерии : учебное пособие / В. Е. Высокогорский, Ю. А. Подольникова, О. Н. Титтель. – Омск : Омский ГАУ, 2025. – 101 с. – ISBN 978-5-907872-46-2. – Текст : электронный // Лань : электронно-библиотечная система. – URL : https://e.lanbook.com/book/482060 . – Режим доступа : для авториз. пользователей	https://e.lanbook.com
Лабковская, Р. Я. Анализ больших данных : учебное пособие / Р. Я. Лабковская, П. В. Косов. – Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2025. – 152 с. – ISBN 978-5-89160-366-0. – Текст : электронный // Лань : электронно-библиотечная система. – URL : https://e.lanbook.com/book/508654 . – Режим доступа: для авториз. пользователей	https://e.lanbook.com
Инженерные технологии и системы : научный журнал. – Саранск : ФГБОУ ВПО "МГУ им. Н.П. Огарёва" – ISSN 2658-6525. – Текст : электронный. – URL : https://znanium.ru/catalog/magazines/issues?ref=eca003ec-77e5-11e9-9e8a-90b11c31de4c . – Режим доступа : по подписке	https://znanium.ru
Программные продукты и системы. – Тверь : Научно-исследовательский институт «Центрпрограммсистем», 1988. – . – Выходит 4 раза в год. – ISSN 2311-2735. – Текст : электронный. – URL : https://eivis.ru/browse/publication/64086	https://eivis.ru/

Пример оформления титульного листа расчетно-аналитической работы

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ П. А. СТОЛЫПИНА»
(ФГБОУ ВО ОМСКИЙ ГАУ)

Экономический факультет

Кафедра экономики, бухгалтерского учета и финансового контроля

ОТЧЕТ
по дисциплине
«Алгоритмы и структуры больших данных»

ОПОП по направлению подготовки 19.04.01 Биотехнология.
Профиль (направленность) «Биоинформатика и анализ данных в
агробiotехнологиях»

**Тема: Оптимизация алгоритма обработки больших
биологических данных**

Номер варианта: 3

Исполнитель: студент группы
... очной формы обучения
ФИО

Проверил: канд. экон. наук, доцент
кафедры ...
ФИО

ОМСК 2027

ПРИЛОЖЕНИЕ 2

Результаты проверки отчета					
№ п/п	Оцениваемая компонента реферата и/или работы над ним	Оценочное заключение преподавателя по данной компоненте			
		Она сформирована на уровне			
		высоком	среднем	минимально приемлемом	ниже приемлемого
1	Соблюдение срока сдачи работы				
2	<i>Оценка содержания отчета</i>				
3	<i>Оценка оформления отчета</i>				
4	<i>Оценка качества подготовки отчета</i>				
5	<i>Оценка ответов на вопросы</i>				
6	Степень самостоятельности обучающегося при подготовке отчета				
Общие выводы и замечания по отчету					
Отчета принят с оценкой:		_____		_____	
		(оценка)		(дата)	
Ведущий преподаватель дисциплины		_____		_____	
		(подпись)		И.О. Фамилия	
Обучающийся		_____		_____	
		(подпись)		И.О. Фамилия	