

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Московский физико-технический институт  
(национальный исследовательский университет)»**

**УТВЕРЖДЕНО**  
**Директор физтех-школы**  
**аэрокосмических технологий**  
**С.С. Негодяев**

	<b>Рабочая программа дисциплины (модуля)</b>
<b>по дисциплине:</b>	Основы параллельного программирования
<b>по направлению:</b>	Прикладные математика и физика
<b>профиль подготовки:</b>	Космические технологии
	Физтех-школа Аэрокосмических Технологий
	кафедра перспективных технологий для систем безопасности
<b>курс:</b>	1
<b>квалификация:</b>	магистр

Семестр, формы промежуточной аттестации: 1 (осенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 0 час.

лабораторные занятия: 30 час.

Самостоятельная работа: 75 час.

Всего часов: 135, всего зач. ед.: 3

Количество контрольных работ, заданий: 2

Программу составил: В.В. Акимов, phd (к.ф.-м.н.), доцент

Программа обсуждена на заседании кафедры перспективных технологий для систем безопасности 31.03.2022

## Аннотация

В настоящем курсе рассматриваются как теоретические проблемы разработки высокоэффективных программ и алгоритмов для современных вычислительных систем, так и прикладные задачи, связанные непосредственно с написанием кода для параллельных вычислений. Основной упор делается на развитие навыков программирования симметричных многопроцессорных, массивно параллельных и распределенных высокопроизводительных систем. Факультативно рассматриваются перспективы развития систем с параллельной архитектурой и средств разработки.

Курс построен на принципе закрепления лекционного материала путем выполнения и разбора домашних практических заданий, которые включают в себя как разработку программного кода, так и оформление результатов в форме научного исследования. Предполагается знакомство с одним из языков программирования, а также базовые знания линейной алгебры, комбинаторики и теории вероятностей.

## 1. Цели и задачи

### Цель дисциплины

- сформировать компетенции в области высокопроизводительных вычислительных систем для их эффективного использования в научно-исследовательской работе;
- выработать навыки решения научно-исследовательских задач с использованием параллельных вычислений;
- развить на практике навыки совместного выполнения научно-исследовательских проектов.

### Задачи дисциплины

- дать основные определения и классификации высокопроизводительных вычислительных систем и методов повышения эффективности решения расчетных задач;
- рассмотреть зависимость эффективности различных подходов к реализации параллельных вычислений от архитектуры используемой вычислительной системы;
- привить навыки программирования вычислительных систем с параллельной архитектурой;
- организовать совместную работу над решением модельной задачи с использованием параллельных вычислений с разделением функций между участниками проекта.

## 2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий	УК-1.1 Анализирует проблемную ситуацию как систему, выявляя ее составляющие и связи между ними
	УК-1.2 Осуществляет поиск вариантов решения поставленной проблемной ситуации на основе доступных источников информации
	УК-1.3 Разрабатывает стратегию достижения поставленной цели как последовательность шагов, предвидя результат каждого из них и оценивая их влияние на внешнее окружение планируемой деятельности и на взаимоотношения участников этой деятельности
УК-2 Способен управлять проектом на всех этапах его реализации	УК-2.1 Формулирует в рамках обозначенной проблемы, цель, задачи, актуальность, значимость (научную, практическую, методическую и иную в зависимости от типа проекта), ожидаемые результаты и возможные сферы их применения
	УК-2.2 Способен прогнозировать результат деятельности и планировать последовательность шагов для достижения данного результата. Формирует план-график реализации проекта в целом и план контроля его выполнения

	УК-2.3 Способен организовать и координировать работу участников проекта, обеспечивать работу команды необходимыми ресурсами
ОПК-1 Владеет системой фундаментальных научных знаний в области физико-математических наук	ОПК-1.2 Способен обобщать и критически оценивать опыт и результаты научных исследований в области профессиональной деятельности
ОПК-2 Имеет представление об актуальных проблемах науки и техники в области своей профессиональной деятельности, способен на научном языке формулировать профессиональные задачи	ОПК-2.1 Имеет представление о современном состоянии исследований в рамках тематической области своей профессиональной деятельности
	ОПК-2.2 Способен оценивать актуальность исследований в области своей профессиональной деятельности и их практическую значимость
	ОПК-2.3 Владеет профессиональной терминологией, используемой в современной научно-технической литературе, обладает навыками устного и письменного изложения результатов научной деятельности в рамках профессиональной коммуникации
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Способен планировать и проводить научные исследования самостоятельно или в составе научного коллектива
ПК-3 Способен профессионально работать с исследовательским и испытательным оборудованием (приборами и установками, специализированными пакетами прикладных программ) в избранной предметной области	ПК-3.1 Понимает принципы работы используемого оборудования (специализированных пакетов прикладных программ)
	ПК-3.2 Способен проводить эксперимент (моделирование) с использованием исследовательского оборудования (пакетов прикладных программ)

### 3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- архитектурные особенности и классификацию многопроцессорных систем;
- классификацию распределенных вычислительных систем;
- основные принципы разработки и реализации параллельных алгоритмов;
- отличительные особенности организации межпроцессорного взаимодействия в зависимости от архитектуры вычислительной системы.

уметь:

- разрабатывать масштабируемые алгоритмы для многопроцессорных систем;
- разрабатывать масштабируемые алгоритмы для распределенных вычислительных систем;
- эффективно реализовывать выбранный алгоритм на языке программирования с поддержкой параллельных вычислений;
- составлять описание алгоритма и выполнять тестирование его работы.

владеть:

- методами и средствами разработки масштабируемых параллельных алгоритмов;
- базовыми навыками программирования многопоточных приложений;
- базовыми навыками программирования распределенных приложений;
- базовыми навыками программирования высокопроизводительных кластерных вычислительных систем;
- навыками самостоятельной и совместной работы над решением научно-исследовательской задачи.

### 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

#### 4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Архитектура и основные классификации параллельных вычислительных систем.	2		1	15
2	Параллельные и последовательные алгоритмы и их графы.	6		7	15
3	Особенности программирования многопоточных приложений.	10		10	15
4	Основы программирования в распределенных вычислительных системах.	8		6	15
5	Вопросы выбора алгоритма для его реализации на вычислительном кластере.	4		6	15
Итого часов		30		30	75
Подготовка к экзамену		0 час.			
Общая трудоёмкость		135 час., 3 зач.ед.			

#### 4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 1 (Осенний)

##### 1. Архитектура и основные классификации параллельных вычислительных систем.

Простое и конвейерное функциональные вычислительные устройства. Приведенная и полная система команд процессора. Оптимизация доступа к данным. Многоуровневая и модульная память. Процессорные матрицы. Многопроцессорные вычислительные системы с общей и распределенной памятью. Микропроцессорные системы. Топологии схем коммутации: кольцо, линейка, решетка, гиперкуб, тор, дерево.

Многопроцессорные, кластерные и сетевые вычислительные системы. Высокопроизводительные вычислительные кластеры и суперкомпьютеры. Анализ архитектуры ЭВМ из списка самых высокопроизводительных в мире и в СНГ.

Систематика Флинна. Потoki данных и потоки команд. Классификация многопроцессорных систем: векторные компьютеры, симметричные мультипроцессорные системы, массивно-параллельные компьютерные системы, кластеры.

Графические процессоры и их использование в вычислительных кластерах.

##### 2. Параллельные и последовательные алгоритмы и их графы.

Абстрактная модель последовательного компьютера и ее влияние на выбор алгоритма. Концепция неограниченного параллелизма. Граф алгоритма: структура и параллельная форма. Абстрактная модель параллельной системы.

Влияние эквивалентных преобразований на устойчивость, число операций и параллелизм вычислений. Принцип сдваивания. Информационное ядро алгоритма. Граф алгоритма и ошибки округления. Оценка параллелизма алгоритма снизу.

Строгие и обобщенные развертки графа алгоритма. Графмашина и параллельные структуры: микро и макропараллелизм. Расщепляющие и линейные развертки для декомпозиции алгоритма. Направленные и регулярные графы. Линейный класс программ. Кусочно-линейные развертки. Прямая и косвенная адресация. Локальные алгоритмы и задача укладки графа. Унификации описания алгоритмов.

### 3. Особенности программирования многопоточных приложений.

Вычислительный процесс и поток (нить). Разделяемые и критичные ресурсы. Особенности организации взаимодействия между вычислительными потоками. Способы работы с критичным ресурсом: режим исключительного доступа и семафоры. Синхронная и асинхронная организация параллельных вычислений.

Технология OpenMP как средство распараллеливания программного кода: формат записи директив препроцессора, область видимости директив, статический и динамический контекст, отделенные директивы. Пульсирующий принцип организации ветвлений. Создание области распараллеливания кода, директива PARALLEL. Распределение вычислительной нагрузки между потоками, директивы: DO, FOR и SECTIONS. Разделяемые и локальные переменные потока, параметры SHARED и PRIVATE директивы PARALLEL. Синхронизация потоков, директивы CRITICAL, ATOMIC и BARRIER. Редукция данных через параметр REDUCTION директивы PARALLEL. Организация доступа к общим переменным, директивы INIT\_LOCK, SET\_LOCK и UNSET\_LOCK.

Особенности реализации многопоточных вычислений в операционной системе MS Windows и Linux.

### 4. Основы программирования в распределенных вычислительных системах.

Репликация кода программы в процессе распределенных вычислений. Организация взаимодействия процессов через интерфейс отправки сообщений MPI. Стандарт спецификаций процедур MPI.

Основные понятия: процессы, сообщения, типы данных, коммуникаторы, топологии. Формат записи команд библиотеки MPI. Структура программы и минимально необходимый набор функций. Команда запуска mpirun.

Функции передачи сообщений MPI\_Send и MPI\_Recv. Описание пересылаемых данных. Идентификация процессов и сообщений. Коллективные операции передачи данных. Эффективная передача данных всем процессам: функция MPI\_Bcast. Операция редукции данных: функция MPI\_Reduce. Синхронизация вычислений: функция MPI\_Barrier. Режим передачи сообщений: блокирующий, синхронный, буферизуемый, по готовности. Общая характеристика дополнительных операций передачи данных.

Базовые типы данных в MPI. Понятие производного типа данных. Карта и сигнатура типа. Методы конструирования производных типов данных: непрерывный, векторный, индексный, упакованный. Общий способ определения производного типа данных.

Понятие группы процессов и коммуникатора. Функции для управления группами и коммуникаторами. Функции и переменные среды выполнения программы MPI. Сравнительная характеристика реализаций библиотеки MPI.

### 5. Вопросы выбора алгоритма для его реализации на вычислительном кластере.

Загруженность, производительность и ускорение как характеристики эффективности выполнения задачи на параллельной вычислительной системе. Учет критичных ресурсов для оценки эффективности вычислительной системы. Законы Амдала и их следствия.

Проблема синхронизации параллельных вычислений. Возможность неоднозначности вычислений в параллельных программах. Проблема взаимоблокировки. Исключение неоднозначности вычислений. Волновые схемы параллельных вычислений. Балансировка вычислительной нагрузки процессоров.

Зависимость выбора оптимального алгоритма от конфигурации кластерной вычислительной системы, нахождение наилучшей топологии вычислительной системы для решения определенной задачи, распараллеливание существующего алгоритма. Декомпозиция алгоритма на параллельно исполняемые блоки вычислений. Распределение заданий по процессорам и балансировка. Организация взаимодействия: синхронизация и взаимоисключение, обмен данными.

## **5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)**

Учебная аудитория для лекционных занятий/лабораторных работ, оснащенная компьютером и мультимедийным оборудованием.

## **6.Перечень рекомендуемой литературы**

### Основная литература

1. Богачев К. Ю. Основы параллельного программирования. - М.: БИНОМ. Лаборатория знаний, 2003
2. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. - СПб.: БХВ–Петербург, 2002
3. Корнеев В.В. Параллельное программирование в MPI. - Москва ; Ижевск: Институт компьютерных исследований, 2003

### Дополнительная литература

1. Гергель В.П., Стронгин Р.Г. Основы параллельных вычислений для многопроцессор-ных вычислительных систем. - Н.Новгород: ННГУ, 2001
2. Немнюгин С., Стесик О. Параллельное программирование для многопроцессорных вычислительных систем. - СПб.: БХВ–Петербург, 2002.

## **7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)**

1. Информационно-аналитические материалы: <http://www.parallel.ru>
5. Информационные материалы по OpenMP: <http://www.openmp.org>
6. Информационные материалы по MPI: <http://www.mpi-forum.org>

## **8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)**

пакеты офисного программного обеспечения Microsoft Office (Word, Excel, PowerPoint, OpenOffice), Acrobat Reader, MPI.

## **9. Методические указания для обучающихся по освоению дисциплины (модуля)**

Успешное освоение курса «Основы параллельного программирования» требует большой самостоятельной работы студента. В программе курса приведено минимально необходимое время для работы студента над темой.

Самостоятельная работа включает в себя:

- чтение и конспектирование рекомендованной литературы;
- проработку учебного материала (по конспектам лекций, семинаров, учебной и научной литературе);
- решение задач, предлагаемых студентам на лекциях;
- подготовку к контрольным работам.

Руководство и контроль за самостоятельной работой студента осуществляется в результате анализа итогов контрольных работ, а также индивидуальных консультаций.

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

**по направлению:** Прикладные математика и физика  
**профиль подготовки:** Космические технологии  
Физтех-школа Аэрокосмических Технологий  
кафедра перспективных технологий для систем безопасности  
**курс:** 1  
**квалификация:** магистр

Семестр, формы промежуточной аттестации: 1 (осенний) - Дифференцированный зачет

**Разработчик:** В.В. Акимов, phd (к.ф.-м.н.), доцент

## 1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий	УК-1.1 Анализирует проблемную ситуацию как систему, выявляя ее составляющие и связи между ними
	УК-1.2 Осуществляет поиск вариантов решения поставленной проблемной ситуации на основе доступных источников информации
	УК-1.3 Разрабатывает стратегию достижения поставленной цели как последовательность шагов, предвидя результат каждого из них и оценивая их влияние на внешнее окружение планируемой деятельности и на взаимоотношения участников этой деятельности
УК-2 Способен управлять проектом на всех этапах его реализации	УК-2.1 Формулирует в рамках обозначенной проблемы, цель, задачи, актуальность, значимость (научную, практическую, методическую и иную в зависимости от типа проекта), ожидаемые результаты и возможные сферы их применения
	УК-2.2 Способен прогнозировать результат деятельности и планировать последовательность шагов для достижения данного результата. Формирует план-график реализации проекта в целом и план контроля его выполнения
	УК-2.3 Способен организовать и координировать работу участников проекта, обеспечивать работу команды необходимыми ресурсами
ОПК-1 Владеет системой фундаментальных научных знаний в области физико-математических наук	ОПК-1.2 Способен обобщать и критически оценивать опыт и результаты научных исследований в области профессиональной деятельности
ОПК-2 Имеет представление об актуальных проблемах науки и техники в области своей профессиональной деятельности, способен на научном языке формулировать профессиональные задачи	ОПК-2.1 Имеет представление о современном состоянии исследований в рамках тематической области своей профессиональной деятельности
	ОПК-2.2 Способен оценивать актуальность исследований в области своей профессиональной деятельности и их практическую значимость
	ОПК-2.3 Владеет профессиональной терминологией, используемой в современной научно-технической литературе, обладает навыками устного и письменного изложения результатов научной деятельности в рамках профессиональной коммуникации
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Способен планировать и проводить научные исследования самостоятельно или в составе научного коллектива
ПК-3 Способен профессионально работать с исследовательским и испытательным оборудованием (приборами и установками, специализированными пакетами прикладных программ) в избранной предметной области	ПК-3.1 Понимает принципы работы используемого оборудования (специализированных пакетов прикладных программ)
	ПК-3.2 Способен проводить эксперимент (моделирование) с использованием исследовательского оборудования (пакетов прикладных программ)



## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Основы параллельного программирования» обучающийся должен:

### знать:

- архитектурные особенности и классификацию многопроцессорных систем;
- классификацию распределенных вычислительных систем;
- основные принципы разработки и реализации параллельных алгоритмов;
- отличительные особенности организации межпроцессорного взаимодействия в зависимости от архитектуры вычислительной системы.

### уметь:

- разрабатывать масштабируемые алгоритмы для многопроцессорных систем;
- разрабатывать масштабируемые алгоритмы для распределенных вычислительных систем;
- эффективно реализовывать выбранный алгоритм на языке программирования с поддержкой параллельных вычислений;
- составлять описание алгоритма и выполнять тестирование его работы.

### владеть:

- методами и средствами разработки масштабируемых параллельных алгоритмов;
- базовыми навыками программирования многопоточных приложений;
- базовыми навыками программирования распределенных приложений;
- базовыми навыками программирования высокопроизводительных кластерных вычислительных систем;
- навыками самостоятельной и совместной работы над решением научно-исследовательской задачи.

## 3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

Текущий контроль успеваемости осуществляется на основе тестирования и опроса студентов на занятиях, а также своевременного выполнения ими учебных заданий.

Примеры контрольных заданий:

1. Разработка алгоритма и его реализация для многопоточных вычислений.
2. Разработка алгоритма и его реализация для распределенных вычислений.
3. Разработка алгоритма и его реализация для массивно-параллельных вычислений.
4. Доклад по выбору алгоритма решения модельной задачи на кластерной системе.
5. Обзорный доклад по теме из двух первых дидактических единиц дисциплины.

## 4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

Промежуточная аттестация по дисциплине проводится в форме дифференцированного зачёта в 9 семестре.

Примерные вопросы для подготовки к дифференцированному зачету;

1. Функциональные вычислительные устройства. Организация доступа к данным.
2. Классификации многопроцессорных вычислительных систем.
3. Микропроцессорные системы. Топологии схем коммутации.
4. Кластерные вычислительные системы и суперкомпьютеры и их классификация.
5. Графические процессоры. Характеристики, архитектура, система команд.
6. Абстрактные модели последовательного компьютера и параллельной системы.
7. Граф алгоритма, его структура и параллельная форма. Концепция неограниченного параллелизма.
8. Информационное ядро алгоритма. Оценка параллелизма алгоритма снизу.
9. Строгие и обобщенные развертки графа алгоритма. Параллельные структуры и граф-машина.

10. Декомпозиция алгоритма. Расщепляющие и линейные развертки. Направленные и регулярные графы. Линейный класс программ.
11. Кусочно-линейные развертки. Прямая и косвенная адресация.
12. Локальные алгоритмы и задача укладки графа. Унификация описания алгоритмов.
13. Вычислительный процесс и поток (нить). Разделяемые и критичные ресурсы.
14. Работа с критичным ресурсом: режим исключительного доступа и семафоры.
15. Синхронная и асинхронная организация параллельных вычислений.
16. Основные функции организации многопоточных вычислений на примере директив компилятору по стандарту OpenMP.
17. Особенности многопоточного программирования в среде Linux и MS Windows.
18. Стандарт MPI как средство организации распределенных вычислений.
19. Средства оптимизации взаимодействия в стандарте MPI. Синхронизация вычислений. Режимы передачи сообщений.
20. Базовые и производные типы данных в MPI. Методы конструирования производных типов.
21. Группы процессов и коммунитаторы, функции управления.
22. Функции и переменные среды выполнения программы MPI. Организация вычислений.
23. Эффективность программного кода: загруженность, производительность, ускорение.
24. Критичные ресурсы. Законы Амдала и их следствия.
25. Проблемы синхронизации, взаимоблокировки, неоднозначности параллельных вычислений.

#### Критерии оценивания

оценка «отлично (10)» выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины при ответе экзаменационного билета и ответе на вопросы по программе дисциплины, а также по результатам контрольных работ;

оценка «отлично (9)» выставляется студенту, показавшему систематизированные, глубокие знания учебной программы дисциплины при ответе экзаменационного билета и ответе на вопросы по программе дисциплины, а также по результатам контрольных работ;

оценка «отлично (8)» выставляется студенту, показавшему систематизированные, знания учебной программы дисциплины при ответе экзаменационного билета и ответе на вопросы по программе дисциплины, а также по результатам контрольных работ;

оценка «хорошо (7)» выставляется студенту по результатам контрольных работ, если он твердо знает материал экзаменационного билета, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

оценка «хорошо (6)» выставляется студенту по результатам контрольных работ, если он знает материал экзаменационного билета, по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе много неточностей;

оценка «хорошо (5)» выставляется студенту по результатам контрольных работ, если он знает материал экзаменационного билета, излагает его, умеет применять полученные знания на практике, не допускает в ответе грубых ошибок;

оценка «удовлетворительно (4)» выставляется студенту по результатам контрольных работ, а также, если во время ответа экзаменационного билета он показал фрагментарный, характер знаний, недостаточно правильные формулировки базовых понятий, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения;

оценка «удовлетворительно (3)» выставляется студенту по результатам контрольных работ, а также, если во время ответа экзаменационного билета он показал разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушение логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

оценка «неудовлетворительно (2-1)» выставляется студенту по результатам контрольных работ, а также, если во время ответа экзаменационного билета, он показал что не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач.

## **5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности**

Дифференцированный зачет по дисциплине «Основы параллельного программирования» проводится путем организации специального опроса в устной форме по вопросам, а также дополнительно заданным экзаменатором.

При проведении устного дифференцированного зачета обучающемуся предоставляется 40 минут на подготовку. Опрос обучающегося по билету на устном зачете не должен превышать одного астрономического часа.

Во время проведения зачета при подготовке ответов на билеты, обучающиеся могут пользоваться программой дисциплины, конспектами лекций, семинаров и любой другой литературой.

Во время проведения зачета при ответе обучающегося на вопросы по билету он не может пользоваться конспектами лекций и семинаров и любой другой литературой.