**Федеральное государственное образовательное   
бюджетное учреждение высшего образования**

**«ФинансовЫЙ УНИВЕРСИТЕТ при   
Правительстве Российской Федерации»**

**(Финансовый университет)**

**Департамент анализа данных и машинного обучения**

**Факультета информационных технологий и анализа больших данных**

УТВЕРЖДАЮ

Проректор по учебной   
 и методической работе

\_\_\_\_\_\_\_\_\_\_ Е.А. Каменева

25.04.2023 г.

**Маковейчук К. А.**

**Технологии параллельного программирования**

**Рабочая программа дисциплины**

для студентов, обучающихся по направлению подготовки

09.03.03 - Прикладная информатика,

ОП «Инженерия данных»,

ОП «Прикладная информатика»,

ОП «Прикладные информационные системы в экономике и финансах»

*Рекомендовано Ученым советом*

*Факультета информационных технологий и анализа больших данных*

(*протокол №31 от 18.04.2023г.*)

*Одобрено Советом учебно-научного*

*Департамента анализа данных и машинного обучения*

*(протокол №2 от 29.03.2023г.)*

**Москва 2023**

СОДЕРЖАНИЕ

[1. Наименование дисциплины 2](#_Toc134211302)

[2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине 2](#_Toc134211303)

[3. Место дисциплины в структуре образовательных программ 5](#_Toc134211304)

[4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся 6](#_Toc134211305)

[5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий 7](#_Toc134211306)

[5.1. Содержание дисциплины 7](#_Toc134211307)

[5.2. Учебно–тематический план 9](#_Toc134211308)

[5.3. Содержание семинаров, практических занятий 12](#_Toc134211309)

[6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине 13](#_Toc134211310)

[6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы 13](#_Toc134211311)

[6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю 14](#_Toc134211312)

[7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине 16](#_Toc134211313)

[8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины 28](#_Toc134211314)

[9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины 29](#_Toc134211315)

[10. Методические указания для обучающихся по освоению дисциплины 31](#_Toc134211316)

[11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем 34](#_Toc134211317)

[12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине 35](#_Toc134211318)

# Наименование дисциплины

«Технологии параллельного программирования».

# 2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

|  |  |  |  |
| --- | --- | --- | --- |
| **Код компетенции** | **Наименование компетенции** | **Индикаторы достижения компетенции** | **Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции** |
| **ОП «Прикладная информатика»** | | | |
| ПКП-3 | Способность при­менять методы раз­работки приложе­ний в сфере эконо­мики и финансов на платформе корпо­ративных информа­ционных систем | 1. Демонстрирует знание назначения и функцио­нал типовых модулей корпоративных инфор­мационных систем, ос­новные методы разра­ботки приложений на их платформе | **Знать** принципы организации и функционирования корпоративных информационных систем с учетом па­раллельных вычислений, основные методы разработки приложений на их платформе с применением техноло­гий параллельного программирова­ния и распределенных вычислений;  **Уметь** применять современные ме­тоды разработки приложений на ос­нове типовых модулей корпоратив­ных информационных систем с ис­пользованием технологий параллель­ного программирования и распреде­ленных вычислений |
| 2. Владеет методологией разработки приложений в сфере экономики и фи­нансов на платформе корпоративных инфор­мационных систем | **Знать** принципы построения архи­тектуры программного обеспечения с учетом параллельных вычислений в сфере экономики и финансов на плат­форме корпоративных информацион­ных систем; стандарты распараллели­вания программ в системах с общей памятью и передачи сообщений в си­стемах с распределенной памятью, директивы, функции, переменные среды; средства разработки для GPU; методы и средства проектирования программного обеспечения для высо­копроизводительных и гетерогенных систем  **Уметь** применять современные под­ходы и методологии из области па­раллельных вычислений и програм­мирования при разработке на плат­форме корпоративных информацион­ных систем приложений в сфере эко­номики и финансов; использовать технологии и существующие типо­вые решения для проектирования программного обеспечения с учетом параллельных вычислений, такие, как OpenMP для систем с общей памя­тью, MPI для систем с распределен­ной памятью, PyOpenCl, PyOpenCuda для распараллеливания процессов на GPU |
| ПКП-5 | Способность при­менять технологии разработки настольных, мо­бильных и web-приложений в сфере экономики и финансов | 1. Демонстрирует знание технологии разработки настольных, мобильных и web- приложений | **Знать** теоретические основы приме­нения технологий параллельного программирования при разработке настольных, мобильных и web- при­ложений в сфере экономики и фи­нансов  **Уметь** применять теоретические ос­новы технологий параллельного про­граммирования при разработке настольных, мобильных и web- при­ложений в сфере экономики и фи­нансов |
| 2. Владеет навыками раз­работки настольных при­ложений в сфере эконо­мики и финансов | **Знать** теоретические основы приме­нения технологий параллельного программирования при разработке настольных приложений в сфере эко­номики и финансов  **Уметь** применять теоретические ос­новы технологий параллельного про­граммирования при разработке настольных приложений в сфере эко­номики и финансов |
| 3. Владеет навыками раз­работки мобильных при­ложений в сфере эконо­мики и финансов | **Знать** теоретические основы приме­нения технологий параллельного программирования при разработке мобильных приложений в сфере эко­номики и финансов  **Уметь** применять теоретические ос­новы технологий параллельного про­граммирования при разработке мо­бильных приложений в сфере эконо­мики и финансов |
| 4. Владеет навыками раз­работки web- приложе­ний в сфере экономики и финансов | **Знать** теоретические основы приме­нения технологий параллельного программирования при разработке web- приложений в сфере экономики и финансов  **Уметь** применять теоретические ос­новы технологий параллельного про­граммирования при разработке web- приложений в сфере экономики и финансов |
| **ОП «Инженерия данных», ОП «Прикладные информационные системы в экономике и финансах»** | | | |
| ПКН-3 | Способность проек­тировать и реализо­вывать архитектуру и дизайн программ­ной системы в соот­ветствии с анали­зом задачи и требо­ваний к ней | 1. Демонстрирует знание основных алгоритмов и структур данных, ис­пользует на практике простые структуры дан­ных, оценивает слож­ность алгоритмов | **Знать** современные подходы, техно­логии и средства разработки для реа­лизации алгоритмов и структур дан­ных методами параллельного про­граммирования  **Уметь** применять современные под­ходы, технологии и средства разра­ботки для реализации алгоритмов и структур данных методами парал­лельного программирования |
| 2. Собирает, формули­рует, систематизирует и анализирует функцио­нальные и нефункцио­нальные требования к ин­формационной системе, выбирает архитектурные решения на их основе | **Знать** принципы организации и функциональные и нефункциональ­ные требования к информационным системам с учетом параллельных вы­числений, принципы выбора архитек­турных решений на их основе с при­менением технологий параллельного программирования и распределенных вычислений;  **Уметь** собирать, формулировать, си­стематизировать и анализировать функциональные и нефункциональ­ные требования к информационным системам с учетом параллельных вы­числений, выбирать архитектурные решения на их основе с использова­нием технологий параллель­ного про­граммирования и распределенных вычислений |
| 3. Создает объектно-ори­ентированный код, ин­капсулирующий условия задачи, производит де­композицию задачи и проектирует систему в пределах одной плат­формы или технологии | **Знать** принципы создания объектно-ориентированного кода с учетом па­раллельных вычислений, декомпози­ции задачи и проектирования си­стемы в пределах одной платформы или технологии; стандарты распарал­лели­вания программ в системах с об­щей памятью и передачи сообщений в си­стемах с распределенной памя­тью, директивы, функции, перемен­ные среды; средства разработки для GPU; методы и средства проектиро­вания программного обеспечения для высо­копроизводительных и гетеро­генных систем  **Уметь** применять современные под­ходы и методологии из области па­раллельных вычислений и програм­мирования при разработке объектно-ориентированного кода, декомпози­ции задачи и проектирования си­стемы в пределах одной платформы или технологии; использовать техно­логии и существующие типовые ре­шения для проектирования про­граммного обеспечения с учетом па­раллельных вычислений, такие, как OpenMP для систем с общей памя­тью, MPI для систем с распределен­ной памятью, PyOpenCl, PyOpenCuda для распараллеливания процессов на GPU |

# 3. Место дисциплины в структуре образовательных программ

Дисциплина «Технологии параллельного программирования» является дисциплиной Цикла профиля (элективный) по направлению подготовки 09.03.03 - Прикладная информатика, ОП «Инженерия данных», ОП «Прикладная информатика», ОП «Прикладные информационные системы в экономике и финансах».

# 

# 4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

**ОП «Прикладная информатика» / ОП «Инженерия данных», ОП «Прикладные информационные системы в экономике и финансах»**

очная форма обучения

|  |  |  |
| --- | --- | --- |
| **Вид учебной работы по дисциплине** | **Всего**  **(в з/е и часах)** | **Семестр 7/6**  **(в часах)** |
| **Общая трудоёмкость дисциплины** | **3/108** | **108** |
| ***Контактная работа -***  ***Аудиторные занятия*** | ***50*** | ***50*** |
| *Лекции* | *16* | *16* |
| *Семинары, практические занятия* | *34* | *34* |
| ***Самостоятельная работа*** | ***58*** | ***58*** |
| Вид текущего контроля | Контрольная работа | Контрольная работа |
| Вид промежуточной аттестации | экзамен / зачет | экзамен / зачет |

**ОП «Прикладная информатика» / ОП «Прикладные информационные системы в экономике и финансах»**

очно-заочная форма обучения

|  |  |  |
| --- | --- | --- |
| **Вид учебной работы по дисциплине** | **Всего**  **(в з/е и часах)** | **Семестр 8 / 7**  **(в часах)** |
| **Общая трудоёмкость дисциплины** | **3/108** | **108** |
| ***Контактная работа -***  ***Аудиторные занятия*** | ***34*** | ***34*** |
| *Лекции* | *16* | *16* |
| *Семинары, практические занятия* | *18* | *18* |
| ***Самостоятельная работа*** | ***74*** | ***74*** |
| Вид текущего контроля | Контрольная работа | Контрольная работа |
| Вид промежуточной аттестации | экзамен / зачет | экзамен / зачет |

# 

**ОП «Прикладные информационные системы в экономике и финансах»**

Институт онлайн-образования, заочная форма обучения

|  |  |  |
| --- | --- | --- |
| **Вид учебной работы по дисциплине** | **Всего**  **(в з/е и часах)** | **Семестр 8**  **(в часах)** |
| **Общая трудоёмкость дисциплины** | **3/108** | **108** |
| ***Контактная работа -***  ***Аудиторные занятия*** | ***12*** | ***12*** |
| *Лекции* | *4* | *4* |
| *Семинары, практические занятия* | *8* | *8* |
| ***Самостоятельная работа*** | ***96*** | ***96*** |
| Вид текущего контроля | Контрольная работа | Контрольная работа |
| Вид промежуточной аттестации | зачет | зачет |

# 5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

# 5.1. Содержание дисциплины

**Раздел 1. Введение в «Технологии параллельного программирования».**

История развития параллелизма в архитектуре ЭВМ. Конвейерная обработка данных и команд. Многофункциональная обработка периферийными процессорами. Векторная обработка. Суперкомпьютеры. Классификация вычислительных систем Флинна. Вычислительные системы с общей и с распределенной памятью. Обзор задач, требующих суперкомпьютеров. Технологии параллелизма. Иерархическая организация памяти в компьютере. Локальность по обращению. Анализ задачи умножения матриц.

**Раздел 2. Основы OpenMP.**

Особенности программирования для систем с общей памятью. Понятие процесса, потока и многопоточности. Технология OpenMP, ее особенности и компоненты. Основные сведения о спецификациях набора стандарта OpenMP. Директивы параллельного программирования. Модели памяти и классы переменных в OpenMp. Режимы выполнения многопоточных программ. Вложенный параллелизм.

**Раздел 3. Директивы распределения и синхронизации работы OpenMP.**

Распараллеливание выполнения циклов. Распределение работы на основе независимых задач. Синхронизация выполнения различных потоков. Распределение операторов одного структурного блока между нитями. Копирование значений приватных переменных.

**Раздел 4. Основы векторизации исполняемого кода в OpenMP.**

Проблемы векторизации. Способы векторизации. Директивы для работы с параллелизмом по данным (векторизация) в OpenMP 4.0.

**Раздел 5. Технологии параллельного программирования для графических процессоров (GPU).**

Возможные преимущества вычислений на графическом процессоре. Задача компьютерной визуализации трехмерных сцен. Архитектура графического процессора (GPU). Средства разработки для графического процессора. Python в работе с GPU. Основы неоднородных (гетерогенных) вычислений. Уровни параллельности в архитектуре GPU. Гетерогенное программирование на PyOpenCL. Построение приложений при помощи PyOpenCL и PyCUDA.

**Раздел 6. Основы технологии MPI.**

Особенности программирования для систем с распределенной памятью. Технология MPI. Основы MPI на Python. Синхронные и асинхронные примитивы взаимодействия. Операции типа точка-точка. Операции коллективного взаимодействия. Параллельный алгоритм умножения матрицы на вектор. Операции сочетания параллельных результатов вычислений. Примитивы синхронизации между процессами. Закон Амдала. Оценки предельно возможного ускорения параллельных алгоритмов.

# 5.2. Учебно–тематический план

**ОП «Инженерия данных», ОП «Прикладная информатика»,**

**ОП «Прикладные информационные системы в экономике и финансах»**

очная форма обучения

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | **Наименование тем (разделов) дисциплины** | **Трудоёмкость в часах** | | | | | **Формы текущего контроля успеваемости** |
| **Всего** | **Контактная работа -**  **Аудиторная работа** | | | **Самостоятельная работа** |
| Общая, в т.ч.: | Лекции | Семинары, практические занятия |
|  | Введение в «Тех­нологии парал­лельного програм­мирования» | 12 | 4 | 2 | 2 | 8 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы OpenMP | 22 | 10 | 2 | 8 | 12 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Директивы рас­пределения и син­хронизации ра­боты OpenMP | 14 | 6 | 2 | 4 | 8 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы векториза­ции исполняемого кода в OpenMP | 14 | 6 | 2 | 4 | 8 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Технологии парал­лельного програм­мирования для графических про­цессоров (GPU) | 24 | 12 | 4 | 8 | 12 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы техноло­гии MPI | 22 | 12 | 4 | 8 | 10 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | В целом по дисциплине | 108 | 50 | 16 | 34 | 58 | Согласно учеб­ному плану: кон­трольная работа |
|  | Итого в % |  | 46 | 32 | 68 | 54 |  |

**ОП «Прикладная информатика»,**

**ОП «Прикладные информационные системы в экономике и финансах»**

очно-заочная форма обучения

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | **Наименование тем (разделов) дисциплины** | **Трудоёмкость в часах** | | | | | **Формы текущего контроля успеваемости** |
| **Всего** | **Контактная работа -**  **Аудиторная работа** | | | **Самостоятельная работа** |
| Общая, в т.ч.: | Лекции | Семинары, практические занятия |
|  | Введение в «Тех­нологии парал­лельного програм­мирования» | 14 | 4 | 2 | 2 | 10 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы OpenMP | 20 | 6 | 2 | 4 | 14 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Директивы рас­пределения и син­хронизации ра­боты OpenMP | 16 | 4 | 2 | 2 | 12 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы векториза­ции исполняемого кода в OpenMP | 16 | 4 | 2 | 2 | 12 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Технологии парал­лельного програм­мирования для графических про­цессоров (GPU) | 22 | 8 | 4 | 4 | 14 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы техноло­гии MPI | 20 | 8 | 4 | 4 | 12 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | В целом по дисциплине | 108 | 34 | 16 | 18 | 74 | Согласно учеб­ному плану: кон­трольная работа |
|  | Итого в % |  | 31 | 47 | 53 | 69 |  |

**ОП «Прикладные информационные системы в экономике и финансах»**

**Институт онлайн-образования, заочная форма обучения**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| №  п/п | **Наименование тем (разделов) дисциплины** | **Трудоёмкость в часах** | | | | | **Формы текущего контроля успеваемости** |
| **Всего** | **Контактная работа-**  **Аудиторная работа** | | | **Самостоятельная работа** |
| Общая, в т.ч.: | Лекции | Семинары, практические занятия |
|  | Введение в «Тех­нологии парал­лельного програм­мирования» | 15 | 1 | 1 | 0 | 14 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы OpenMP | 20 | 2 | 1 | 1 | 18 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Директивы рас­пределения и син­хронизации ра­боты OpenMP | 19 | 3 | 1 | 2 | 16 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы векториза­ции исполняемого кода в OpenMP | 15 | 1 | 0 | 1 | 14 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Технологии парал­лельного програм­мирования для графических про­цессоров (GPU) | 21 | 3 | 1 | 2 | 18 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | Основы техноло­гии MPI | 18 | 2 | 0 | 2 | 16 | Самостоятельные работы. Участие в решении задач на практических заня­тиях. Обсуждение решенных задач. |
|  | В целом по дисциплине | 108 | 12 | 4 | 8 | 96 | Согласно учеб­ному плану: кон­трольная работа |
|  | Итого в % |  | 11 | 33 | 67 | 89 |  |

# 

# 5.3. Содержание семинаров, практических занятий

|  |  |  |
| --- | --- | --- |
| **Наименование тем (разделов) дисциплины** | **Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9 (указывается раздел и порядковый номер источника)** | **Формы проведения занятий** |
| Введение в «Тех­нологии парал­лельного програм­мирования» | Суперкомпьютеры.  Классификация вычислительных систем Флинна.  Технологии параллелизма.  Иерархическая организация памяти в компь­ютере.  Локальность по обращению.  Анализ задачи умножения матриц.  *Рекомендуемые источники: 8. 1-4* | Практикум по реше­нию задач по тема­тике занятия и коллек­тивное обсуждение решений  Дискуссия «Супер­компьютеры – ТОП 100: анализ» |
| Основы OpenMP | Основные сведения о спецификациях набора стандарта OpenMP.  Директивы параллельного программирова­ния.  Модели памяти и классы переменных в OpenMp.  Режимы выполнения многопоточных про­грамм.  Вложенный параллелизм.  *Рекомендуемые источники: 8. 1-4* | Практикум по реше­нию задач по тема­тике занятия и коллек­тивное обсуждение решений  Выполнение тестовых заданий |
| Директивы рас­пределе­ния и син­хронизации ра­боты OpenMP | Распараллеливание выполнения циклов.  Распределение работы на основе независи­мых задач.  Синхронизация выполнения различных пото­ков.  Распределение операторов одного структур­ного блока между нитями.  Копирование значений приватных перемен­ных.  *Рекомендуемые источники: 8. 1-4* | Практикум по реше­нию задач по тема­тике занятия и коллек­тивное обсуждение решений  Выполнение тестовых заданий |
| Основы векториза­ции ис­полняемого кода в OpenMP | Директивы для работы с параллелизмом по данным (векторизация) в OpenMP 4.0.  *Рекомендуемые источники: 8. 1-4* | Практикум по реше­нию задач по тема­тике занятия и коллек­тивное обсуждение решений |
| Технологии парал­лель­ного програм­мирования для графических про­цес­соров (GPU) | Python в работе с GPU.  Основы неоднородных (гетерогенных) вы­числений.  Уровни параллельности в архитектуре GPU.  Гетерогенное программирование на PyOpenCL.  Построение приложений при помощи PyOpenCL и PyCUDA.  *Рекомендуемые источники: 8. 1-4* | Практикум по реше­нию задач по тема­тике занятия и коллек­тивное обсуждение решений  Выполнение тестовых заданий |
| Основы техноло­гии MPI | Основы MPI на Python.  Синхронные и асинхронные примитивы взаи­модействия.  Операции типа точка-точка.  Операции коллективного взаимодействия.  Параллельный алгоритм умножения матрицы на вектор.  Операции сочетания параллельных результа­тов вычислений.  Примитивы синхронизации между процес­сами.  *Рекомендуемые источники: 8. 1-4* | Практикум по реше­нию задач по тема­тике занятия и коллек­тивное обсуждение решений  Выполнение тестовых и контрольных зада­ний |

# 6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

# 6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

|  |  |  |
| --- | --- | --- |
| **Наименование тем (разделов) дисциплины** | **Перечень вопросов, отводимых на самостоятельное освоение** | **Формы внеаудиторной самостоятельной работы** |
| Введение в «Тех­нологии парал­лельного програм­мирования» | История развития параллелизма в архи­тектуре ЭВМ.  Конвейерная обработка данных и ко­манд.  Многофункциональная обработка пери­ферийными процессорами.  Векторная обработка.  Вычислительные системы с общей и с распределенной памятью.  Обзор задач, требующих суперкомпью­теров. | Работа с текстом лекции, разбор вопросов и заданий по теме занятия;  изучение рекомендован­ных к занятию литератур­ных источников. |
| Основы OpenMP | Особенности программирования для си­стем с общей памятью.  Понятие процесса, потока и многопо­точности.  Технология OpenMP, ее особенности и компоненты. | Работа с текстом лекции, разбор вопросов и заданий по теме занятия;  изучение рекомендован­ных к занятию литератур­ных источников. |
| Директивы рас­пределения и син­хронизации ра­боты OpenMP | Примеры задач, допускающих распарал­леливание выполнения циклов.  Сравнение эффективности распаралле­ливания циклов (директива For) и рас­пределения работы на основе независи­мых задач (директива Task) на примере различных задач. | Работа с текстом лекции, разбор вопросов и заданий по теме занятия;  изучение рекомендован­ных к занятию литератур­ных источников. |
| Основы векториза­ции исполняемого кода в OpenMP | Проблемы векторизации.  Способы векторизации. | Работа с текстом лекции, разбор вопросов и заданий по теме занятия;  изучение рекомендован­ных к занятию литератур­ных источников. |
| Технологии парал­лельного програм­мирования для графических про­цессоров (GPU) | Возможные преимущества вычислений на графическом процессоре.  Задача компьютерной визуализации трехмерных сцен.  Архитектура графического процессора (GPU).  Средства разработки для графического процессора. | Работа с текстом лекции, разбор вопросов и заданий по теме занятия;  изучение рекомендован­ных к занятию литератур­ных источников. |
| Основы техноло­гии MPI | Особенности программирования для си­стем с распределенной памятью.  Технология MPI.  Закон Амдала.  Оценки предельно возможного ускоре­ния параллельных алгоритмов. | Работа с текстом лекции, разбор вопросов и заданий по теме занятия;  изучение рекомендован­ных к занятию литератур­ных источников. |

# 6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

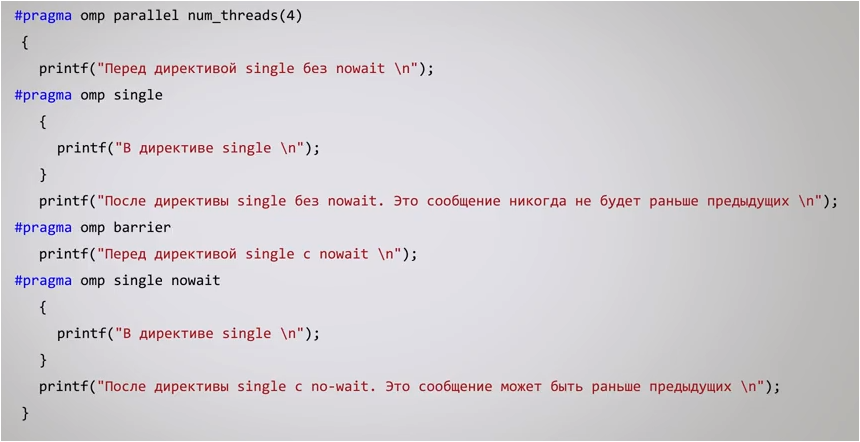
**Примерные задания контрольной работы**

**Задание 1.** После выполнения выделенного параллельного участка в программе происходит неявная барьерная синхронизация параллельно работающих потоков. Их дальнейшее выполнение происходит только тогда, когда все они достигнут данной точки. Если в подобной задержке нет необходимости, опция nowait позволяет потокам, уже дошедшим до конца участка, продолжить выполнение без синхронизации с остальными. Реализуйте программу, иллюстрирующую описанную ситуацию.

Реализуйте программу на C++ с подключением технологии OpenMP, и проверьте получаемый результат, обоснуйте результат в выводах.

Можно использовать компилятор g++ совместно с установленной для него реализацией технологии OpenMP, либо IDE MS Visual Studio Community с настройками, разрешающими OpenMP, в свойствах проекта, либо использовать любой другой доступный компилятор С++ с OpenMP, в том числе облачный, например, <https://www.codingame.com/playgrounds/54443/openmp/playground>

Пример реализации:



**Задание 2.** Выполнить умножение матриц, используя любые изученные технологии распараллеливания и ускорения вычислений (OpenMP, Mpi4py, PyOpenCL). Сравнить производительность программы для матриц разного размера и при разном количестве потоков (для OpenMP); рабочих процессов (для MPI), для CPU и GPU (для PyOpenCL).

*Критерии балльной оценки различных форм текущего контроля успеваемости содержатся в соответствующих методических рекомендациях Департамента анализа данных и машинного обучения.*

# 7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Перечень компетенций, формируемых в процессе освоения дисциплины, содержится в разделе **2 «Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».**

**Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний**

|  |  |  |  |
| --- | --- | --- | --- |
| **Наименование компетенции** | **Индикаторы до­стижения компе­тенции** | **Результаты обучения (умения и знания), соотнесенные с ин­дикаторами достижения ком­петенции** | **Типовые контроль­ные задания** |
| **ОП «Прикладная информатика»** | | | |
| ПКП-3. Спо­собность при­менять методы раз­работки приложе­ний в сфере эконо­мики и финан­сов на плат­форме корпо­ративных ин­форма­ционных систем | 1. Демонстрирует знание назначения и функцио­нал ти­повых модулей корпоративных инфор­мационных систем, ос­новные методы разра­ботки приложений на их платформе | **Знать** принципы организа­ции и функционирования корпоратив­ных информа­ционных систем с учетом па­раллельных вычисле­ний, основные методы разра­ботки приложений на их плат­форме с применением техноло­гий параллельного программи­рова­ния и рас­пределенных вы­числений;  **Уметь** применять совре­менные ме­тоды разработки приложений на ос­нове ти­повых модулей кор­поратив­ных информационных си­стем с ис­пользованием тех­но­логий параллель­ного програм­мирования и рас­преде­ленных вычислений | Выполнить обследо­ва­ние задач финансо­вого предприятия «Зета» в соот­ветствии принципами орга­низа­ции информационных систем. Оценить задачи с точки зре­ния приме­нимо­сти алгорит­мов параллель­ных вычисле­ний. |
| 2. Владеет методо­логией разработки приложений в сфере экономики и фи­нансов на платформе корпо­ративных инфор­мационных си­стем | **Знать** принципы построе­ния архи­тектуры программ­ного обеспечения с учетом парал­лельных вычислений в сфере экономики и финан­сов на плат­форме корпора­тивных инфор­мацион­ных систем; стандарты распа­раллели­вания программ в системах с общей памятью и пе­редачи сообщений в си­стемах с распределенной памятью, ди­рективы, функ­ции, переменные среды; средства разработки для GPU; методы и средства проек­тирования программ­ного обес­печения для высо­копроизводи­тельных и ге­терогенных систем  **Уметь** применять совре­менные под­ходы и методо­логии из об­ласти па­рал­лельных вычисле­ний и про­грам­мирования при разра­ботке на плат­форме корпо­ративных информацион­ных си­стем приложений в сфере эко­номики и финансов; ис­пользо­вать технологии и существую­щие типо­вые ре­шения для про­ектирования программного обеспечения с учетом парал­лельных вы­числений, такие, как OpenMP для систем с об­щей па­мя­тью, MPI для си­стем с рас­пределен­ной па­мятью, Py­OpenCl, PyOpen­Cuda для распа­раллелива­ния процессов на GPU | Реализовать поста­новку задачи, алгоритм и программу решения задачи обработки условных финансовых данных с помощью ите­рационного ме­тода ре­шения системы линей­ных алгебраических уравнений (СЛАУ) как умножение матрицы на век­тор с использова­нием тех­нологий па­раллельного програм­мирования и рас­преде­ленных вычислений (возможные техноло­гии для реализации программы: С++ & OpenMP | Python + PyOpenCL | Python + mpi4py) |
| ПКП-5. Спо­собность при­менять техно­логии разра­ботки настоль­ных, мо­биль­ных и web-приложений в сфере эконо­мики и финан­сов | 1. Демонстрирует знание технологии разработки настольных, мо­бильных и web- приложений | **Знать** теоретические ос­новы приме­нения техноло­гий парал­лельного про­граммирования при разра­ботке настольных, мо­биль­ных и web- при­ложений в сфере экономики и фи­нан­сов  **Уметь** применять теорети­че­ские ос­новы технологий парал­лельного про­грамми­рования при разработке настольных, мо­бильных и web- при­ложений в сфере экономики и фи­нансов | Определить инстру­мен­тальные средства информа­ционных тех­нологий для реализа­ции программы по­стро­ения фрактала Нью­тона для уравнения х3-1=0  Обосновать примене­ние технологий парал­лельного программиро­вания при раз­работке приложения. |
| 2. Владеет навы­ками раз­работки настольных при­ложений в сфере эконо­мики и фи­нансов | **Знать** теоретические ос­новы приме­нения техноло­гий парал­лельного про­граммирования при разра­ботке настольных при­ложе­ний в сфере эко­номики и финансов  **Уметь** применять теорети­че­ские ос­новы технологий парал­лельного про­грамми­рования при разработке настольных при­ложений в сфере эко­номики и финан­сов | Реализовать поста­новку за­дачи, алгоритм и настоль­ное приложе­ние для реше­ния задачи по­строения фрактала Нью­тона для уравнения х3-1=0  Применить в решении за­дачи технологии па­раллель­ного програм­мирования. |
| 3. Владеет навы­ками раз­работки мобильных при­ло­жений в сфере эко­но­мики и финан­сов | **Знать** теоретические ос­новы приме­нения техноло­гий парал­лельного про­граммирования при разра­ботке мобильных при­ложе­ний в сфере эко­номики и финансов  **Уметь** применять теорети­че­ские ос­новы технологий парал­лельного про­грамми­рования при разработке мо­бильных при­ложений в сфере эконо­мики и финан­сов | Реализовать поста­новку за­дачи, алгоритм и мобильное приложе­ние для решения задачи по­строения фрак­тала Нью­тона для уравне­ния х3-1=0  Применить в решении за­дачи технологии па­раллель­ного програм­мирования. |
| 4. Владеет навы­ками раз­работки web- приложе­ний в сфере экономики и финансов | **Знать** теоретические ос­новы приме­нения техноло­гий парал­лельного про­граммирования при разра­ботке web- приложе­ний в сфере экономики и финан­сов  **Уметь** применять теорети­че­ские ос­новы технологий парал­лельного про­грамми­рования при разработке web- приложе­ний в сфере экономики и финан­сов | Реализовать поста­новку за­дачи, алгоритм и web-приложение для решения задачи по­строения фрак­тала Нью­тона для уравне­ния х3-1=0  Применить в решении за­дачи технологии па­раллель­ного програм­мирования. |
| **ОП «Инженерия данных», ОП «Прикладные информационные системы в экономике и финансах»** | | | |
| ПКН-3. Спо­собность про­ек­тировать и реализо­вывать архитектуру и дизайн про­грамм­ной си­стемы в соот­ветствии с ана­ли­зом задачи и требо­ваний к ней | 1. Демонстри­рует знание ос­новных алгорит­мов и структур данных, ис­поль­зует на практике простые струк­туры дан­ных, оценивает слож­ность алгорит­мов | **Знать** современные под­ходы, техно­логии и средства разработки для реа­лизации алгоритмов и структур дан­ных методами параллель­ного про­граммирования  **Уметь** применять современ­ные под­ходы, технологии и средства разра­ботки для реа­лизации алгоритмов и струк­тур данных методами парал­лельного программирования | Реализовать структуру дан­ных и алгоритм умножения векторов в Python, с исполь­зованием GPU (библиотека PyOpenCL или PyOpen­Cuda), либо с использова­нием библиотеки mpi4py. Сравнить два метода парал­лельного программирова­ния. |
| 2. Собирает, формули­рует, систематизирует и анализирует функцио­наль­ные и нефункци­о­нальные требо­вания к ин­фор­мационной си­стеме, выбирает архитектурные решения на их основе | **Знать** принципы организа­ции и функциональные и не­функциональ­ные требова­ния к информационным си­стемам с учетом параллель­ных вы­числений, принципы выбора архитек­турных ре­шений на их основе с при­ме­нением технологий парал­лельного программирования и распределенных вычисле­ний;  **Уметь** собирать, формули­ровать, си­стематизировать и анализировать функцио­нальные и нефункциональ­ные требования к информа­ционным системам с учетом параллельных вы­числений, выбирать архитектурные ре­шения на их основе с исполь­зова­нием технологий парал­лель­ного про­граммирования и распределенных вычисле­ний | Определить спецификацию функциональных и нефунк­циоанльных требований к приложению для расчета прогноза погоды на основа­нии 5 матриц исходных дан­ных для точек атмосферы на 4 высотах, с применением технологий параллельного программирования и распре­деленных вычислений. |
| 3. Создает объ­ектно-ори­енти­рованный код, ин­капсулирую­щий условия за­дачи, произво­дит де­компози­цию задачи и проектирует си­стему в преде­лах одной плат­формы или тех­нологии | **Знать** принципы создания объектно-ориентированного кода с учетом па­раллельных вычислений, декомпози­ции задачи и проектирования си­стемы в пределах одной платформы или технологии; стандарты распарал­лели­ва­ния программ в системах с об­щей памятью и передачи сообщений в си­стемах с рас­пределенной памя­тью, ди­рективы, функции, перемен­ные среды; средства разра­ботки для GPU; методы и средства проектиро­вания программного обеспечения для высо­копроизводитель­ных и гетеро­генных систем  **Уметь** применять современ­ные под­ходы и методологии из области па­раллельных вычислений и програм­миро­вания при разработке объ­ектно-ориентированного кода, декомпози­ции задачи и проектирования си­стемы в пределах одной платформы или технологии; использо­вать техно­логии и существу­ющие типовые ре­шения для проектирования про­грамм­ного обеспечения с учетом па­раллельных вычислений, такие, как OpenMP для си­стем с общей памя­тью, MPI для систем с распределен­ной памятью, PyOpenCl, PyOpen­Cuda для распараллеливания процессов на GPU | Разработать приложение для расчета прогноза погоды на основании 5 матриц исход­ных данных для точек атмо­сферы на 4 высотах, с приме­нением технологий парал­лельного программирования и распределенных вычисле­ний. Прогноз условно вы­полняется на основании ре­шения уравнения 5х5+3х4-7х3+2х2-8х+15=у, где х – данные матриц по каждой высоте.  Выполнить разработку с применением па­раллельных вычислений и выбранной технологии, такой, как OpenMP, MPI, PyOpenCl или PyOpen­Cuda для распаралле­ливания процессов на GPU |

**Примеры тестовых заданий**

**1. Какие подходы, применяемые в архитектурах компьютеров, позволяли повысить их производительность**

-Конвейерный способ обработки

-Независимые функциональные устройства

-Векторные операции верно

-Уменьшение энергопотребления

**2. Какой принцип организации памяти используется для обеспечения требуемой емкости и высокого быстродействия памяти компьютера за приемлемую цену**

-Интенсивный

-Экстенсивный

-Масштабируемости

-Иерархический

**3. Самый минимальный по объему тип памяти, но и самый быстрый это**

-Регистры центрального процессора

-Кэш первого уровня

-Оперативная память

-Кэш нулевого уровня

**4. Для эффективного использования иерархической структуры памяти при написании программ необходимо использовать**

- Принцип локальность по обращению

- Принципы объектного-ориентированного программирования

- Принцип случайного обращения

- Принцип маленьких данных

**5. Классификация Флинна основывается на**

-на понятии потока

- на понятии декомпозиции данных

-на особенностях процессоров

-на особенностях организации памяти

**6. SISD это**

- одиночный поток команд и одиночный поток данных

- множественный поток команд и одиночный поток данных

- одиночный поток команд и множественный поток данных

- множественный поток команд и множественный поток данных

**7. Какие два крупных класса параллельных высокопроизводительных вычислительных систем можно выделить?**

-вычислительные системы с общей памятью

-вычислительные системы с распределенной памятью

-вычислительные системы с очень большим объемом оперативной памяти

- вычислительные системы с водяным охлаждением

**8. Какие из этих технологий позволяют создавать параллельные программы**

- OpenMP

- MPI

-Pascal

-Parallel

**9. Многопоточность это**

- свойство платформы или приложения, состоящее в том, что процесс, порождённый в операционной системе, может состоять из нескольких потоков, выполняющихся «параллельно», то есть без предписанного порядка во времени.

- свойство центрального процессора автоматически выполнять программы сразу в несколько потоков

- свойство компилятора позволяющие создавать параллельные многопотоковые программы.

**10. Технология параллельного программирования OpenMP включается в себя**

-директивы компилятора

-библиотечные функции

-переменные среды окружения

-параллельный компилятор

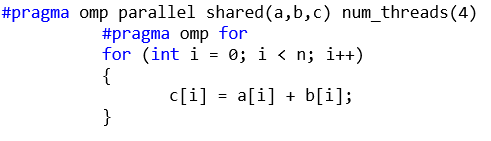
**11. Для создания параллельных программ в OpenMP применяется модель программирования**

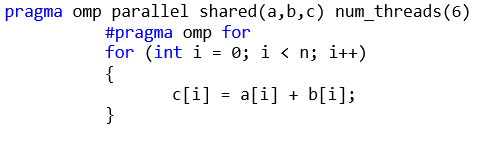
-Разветвление-слияние (Fork–join)

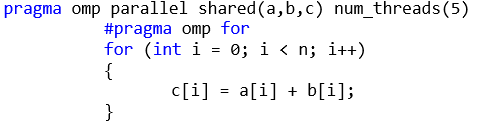
-Декомпозиция по данным

-Главный подчиненный (Master-slave)

**12. Выберите код в котором создается параллельная область с 5 дополнительными потоками выполнения**

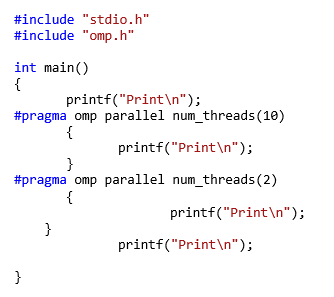




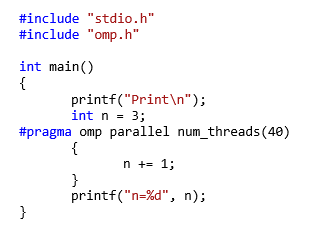


**13. Следующий фрагмент кода запустили на компьютере с 8 ядерным процессором**

**Сколько раз будет выведено слово Print?**



**14. Какое значение переменной n будет выведено на экран в следующей программе при запуске на компьютере с 4-х ядерным процессором?**



Значение не определенно, от запуска к запуску может меняться

Программа не запустится, так как потоков больше чем ядер

Значение 43 будет выведено

Программа не скомпилируется из-за синтаксической ошибки

**15. Выберите два основных класса переменных в OpenMP**

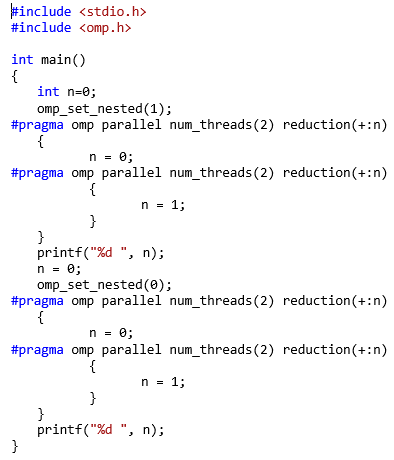
-общие

-локальные

-глобальные

-независимые

**16. В результате выполнения следующей программы на компьютере с 4 ядерным процессором на экран будут выведены числа. Впишите каждое из чисел в отдельное окно ввода**



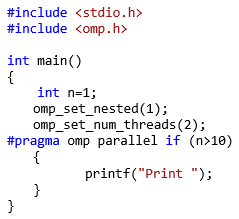
**17. Если несколько потоков одновременно записывают значение общей переменной без выполнения синхронизации, то**

-возникает ситуации «гонки данных» (data race)

-возникает ситуации не разделенного доступа

-в данном случаи нет никаких особенных ситуаций

**18. В результате выполнения данной программы на компьютере с восьмиядерным процессором слов «Print» будет выведено (ввести числом)**



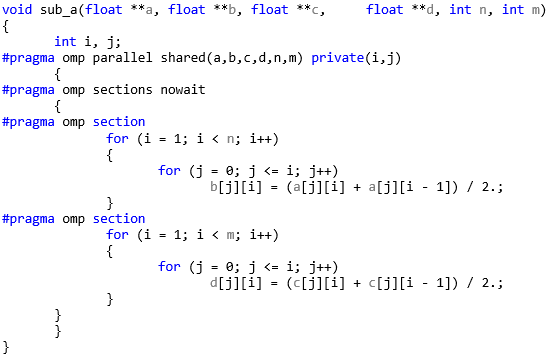
**19. Для выделения независимой задачи, которая может исполняться параллельно используется директива**

#pragma omp task

#pragma omp section

#pragma omp doit

**20. Какое максимальное ускорение (ускорение = время на 1 ядре / время на n ядрах ) возможно в следующей программе при запуске на компьютере с 8 ядерным процессором**



Максимальное ускорение зависит от количества используемых ядер

Максимальное ускорение 8

Максимальное ускорение 2

**Примерные вопросы для подготовки к зачету**

1. Что такое процесс и поток в современных операционных системах.

2. Для чего необходима синхронизация в многопоточных программах.

3. В каких состояниях может находиться поток.

4. В результате каких действий поток может переходить из одного состояния в другое.

5. Когда необходимо применять синхронизованные методы.

6. Какие примитивы позволяют осуществить обмен сообщениями между процессами.

7. Примитивы коллективного взаимодействия процессов.

8. Примитивы синхронизации между процессами.

9. Директивы распределения работы и синхронизации работы.

10. Распараллеливание выполнения циклов.

11. Распределение нескольких структурных блоков между потоками

12. Распределение работы на основе независимых задач

13. Синхронизация выполнения различных потоков. Простые директивы

14. Векторные вычисления с помощью OpenMP 4.0

15. Анализ и оптимизация программ с использованием современных компиляторов

16. Архитектура графического процессора (GPU).

17. Возможные преимущества вычислений на графическом процессоре.

18. Средства разработки для графического процессора.

19. Закон Амдала.

20. Классификация вычислительных систем Флинна.

**Примерные вопросы для подготовки к экзамену**

1. История появления параллельной и высокопроизводительной обработки информации. Суперкомпьютеры.
2. Классификация вычислительных систем Флинна. Примеры.
3. Технологии параллелизма. Классификация, примеры.
4. Принцип иерархической организации памяти в компьютере. Примеры.
5. Локальность по обращению. Принципы локальности памяти.
6. Анализ задачи умножения матриц.
7. Основные сведения о спецификациях набора стандарта OpenMP.
8. Директивы параллельного программирования OpenMP.
9. Модели памяти и классы переменных в OpenMp.
10. Режимы выполнения многопоточных программ.
11. Вложенный параллелизм в OpenMP.
12. Директивы распределения и синхронизации работы OpenMP
13. Распараллеливание выполнения циклов в OpenMP: директива и принцип организации.
14. Распределение работы на основе независимых задач в OpenMP: директива и принцип организации.
15. Синхронизация выполнения различных потоков.
16. Распределение операторов одного структурного блока между нитями.
17. Копирование значений приватных переменных.
18. Основы векторизации исполняемого кода в OpenMP
19. Директивы для работы с параллелизмом по данным (векторизация) в OpenMP 4.0.
20. Технологии параллельного программирования для графических процессоров (GPU)
21. Python в работе с GPU: реализации библиотек, сравнение.
22. Основы неоднородных (гетерогенных) вычислений.
23. Уровни параллельности в архитектуре GPU.
24. Гетерогенное программирование на PyOpenCL и PyCUDA.
25. Построение приложений при помощи PyOpenCL и PyCUDA.
26. Основы технологии MPI
27. Основы MPI для Python.
28. Синхронные и асинхронные примитивы взаимодействия MPI.
29. Операции типа точка-точка в MPI.
30. Операции коллективного взаимодействия в MPI.
31. Параллельный алгоритм умножения матрицы на вектор: реализации на MPI.
32. Операции сочетания параллельных результатов вычислений в MPI.
33. Примитивы синхронизации между процессами в MPI.

**Пример экзаменационного билета**

**Задание 1.** Теоретический вопрос. Директивы распределения и синхронизации работы OpenMP ***(20 баллов)***

**Задание 2.** Практический вопрос. Реализовать умножение векторов в Python, написать две функции реализации: для CPU и GPU (библиотека PyOpenCL). Сравнить время выполнения CPU, GPU, kernel. ***(20 баллов)***

**Задание 3. Практический вопрос.** Реализовать функцию умножения векторов в Python, с использованием библиотеки mpi4py. Сравнить время выполнения функции с временем выполнения функций из задания 2. ***(20 баллов)***

# 8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

**а) основная:**

1. Малявко, А. А. Параллельное программирование на основе технологий openmp, mpi, cuda : учебное пособие для академического бакалавриата / А. А. Малявко. — 2-е изд., испр. и доп. — Москва : Юрайт, 2023. — 129 с. —— ЭБС Юрайт. — URL: https://biblio-online.ru/bcode/514199 (дата обращения: 19.05.2023). — Текст : электронный.
2. Арыков, С. Б. Параллельное программирование над общей памятью. OpenMP : учебное пособие / С. Б. Арыков, М. А. Городничев, Г. А. Щукин. — Новосибирск : НГТУ, 2019. — 95 с. — ЭБС Лань. — URL: https://e.lanbook.com/book/152252 (дата обращения: 19.05.2023). — Текст : электронный.

**б) дополнительная:**

1. Карепова, Е. Д. Основы многопоточного и параллельного программирования: учебное пособие / Е. Д. Карепова. – Красноярск : Сиб. Федер. ун-т, 2016. — 356 с. — ЭБС ZNANIUM.com. — URL: http://znanium.com/catalog/product/966962 (дата обращения: 19.05.2023). — Текст : электронный.
2. Филатов, А. С. Параллельное программирование : учебное пособие / А. С. Филатов. — Москва : РТУ МИРЭА, 2021. — 46 с. — ЭБС Лань. — URL: https://e.lanbook.com/book/218429 (дата обращения: 19.05.2023). — Текст : электронный.

# 9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. <http://www.fa.ru> – официальный сайт Финансового университета при Правительстве Российской Федерации
2. <http://www.library.fa.ru/> - Библиотечно-информационный комплекс Финуниверситета
3. Электронно-библиотечная система BOOK.RU http://www.book.ru
4. Электронно-библиотечная система «Университетская библиотека ОНЛАЙН» http://biblioclub.ru/
5. Электронно-библиотечная система Znanium http://www.znanium.com
6. Электронно-библиотечная система издательства «ЮРАЙТ» https://urait.ru/
7. Электронно-библиотечная система издательства Проспект http://ebs.prospekt.org/books
8. Электронно-библиотечная система издательства Лань https://e.lanbook.com/
9. Деловая онлайн-библиотека Alpina Digital http://lib.alpinadigital.ru/
10. Электронная библиотека Издательского дома «Гребенников» https://grebennikon.ru/
11. Научная электронная библиотека eLibrary.ru http://elibrary.ru
12. Национальная электронная библиотека http://нэб.рф/
13. [https://org.fa.ru](https://org.fa.ru/) - личный кабинет обучающегося
14. <https://org.fa.ru/app/ebs/list> - реестр ЭБС Финансового университета
15. <https://openedu.ru/course/msu/PARPROG/> - онлайн-курс «Суперкомпьютеры и параллельная обработка данных» на национальной платформе «Открытое образование» (курс МГУ им. М. В, Ломоносова)
16. <https://mooc.tsu.ru/mooc-openedu/mpi/> - онлайн-курс «Параллельное программирование с использованием OpenMP и MPI» на платформе МООК Томского государственного университета
17. <https://stepik.org/course/115024/promo?search=2147452613> - онлайн-курс «Введение в параллельное программирование (OpenMP и MPI)» на платформе Stepik (Томский государственный университет, альтернативная платформа)
18. <https://teach-in.ru/course/parallel-computing-lukyanenko/lecture> - онлайн-курс «Параллельные вычисления» на платформе «Лекторий Teach-in» (лекции ученых МГУ им. М. В, Ломоносова)
19. <https://campus.fa.ru/course/view.php?id=15769> – базовый ЭУК по дисциплине «Технологии параллельного программирования» Финансового университета
20. <http://onreader.mdl.ru/PythonParallelProgrammingCookbook.2nd/content/index.html> - Закконе Джанкарло. Книга рецептов параллельного программирования Python. 2е изд. Copyright © 2019 Packt Publishing. Документ предоставляется по лицензии Creative Commons Attribution 3.0 License.
21. <https://registry.khronos.org/OpenCL/specs/3.0-unified/html/OpenCL_API.html> - The OpenCL™ Specification Khronos® OpenCL Working Group Version V3.0.14, Mon, 17 Apr 2023
22. <https://github.com/inducer> - Andreas Klöckner's web page PyOpenCL
23. <https://documen.tician.de/pyopencl/> - PyOpenCL’s documentation
24. <https://pypi.org/project/mpi4py/> - Python bindings for MPI
25. <https://mpi4py.readthedocs.io/en/stable/index.html> - MPI for Python

# 10. Методические указания для обучающихся по освоению дисциплины

Цель методических рекомендаций – обеспечить студенту бакалавриата (далее – студенту) оптимальную организацию процесса изучения дисциплины, а также выполнения различных форм самостоятельной работы.

***Методические рекомендации по изучению дисциплины***

Студентам необходимо ознакомиться содержанием рабочей программы дисциплины (далее – РПД), с целями и задачами дисциплины, ее связями с другими дисциплинами образовательной программы, списком рекомендованной литературы и перечнем ресурсов сети «Интернет», необходимыми для освоения дисциплины. Часть разделов дисциплины студенты могут освоить с помощью бесплатных онлайн-курсов (с получением сертификата или иного документа об окончании), указанных в перечне ресурсов. Лекции, задания для семинарских и самостоятельных занятий, презентации и другие материалы по дисциплине, представленные в базовом ЭУК по дисциплине на образовательном портале Финансового университета, студенты изучают в процессе освоения курса в течение семестра. На сайте департамента студенты могут ознакомиться в течение семестра с графиком консультаций преподавателей.

***Рекомендации по подготовке к лекционным занятиям***

***(теоретический курс)***

Изучение дисциплины требует систематического и последовательного накопления знаний, следовательно, пропуски отдельных тем не позволяют глубоко освоить предмет. Именно поэтому контроль над систематической работой студентов всегда находится в центре внимания департамента.

Лекционные занятия проводятся в соответствии с тематическим планом, при изложении материала рекомендуется использовать презентации в среде PowerPoint и фрагменты печатных материалов по теме лекции.

Студентам рекомендуется:

- перед каждой лекцией просматривать рабочую программу дисциплины, что позволит сэкономить время на записывание темы лекции, ее основных вопросов, рекомендуемой литературы;

- на отдельные лекции приносить соответствующий материал на бумажных или электронных носителях, представленный лектором на портале. Данный материал будет охарактеризован, прокомментирован, дополнен непосредственно не лекции;

- перед очередной лекцией необходимо просмотреть по конспекту материал предыдущей лекции. При затруднениях в восприятии материала следует обратиться к основным литературным источникам, если разобраться в материале опять не удалось, то обратитесь к лектору (по графику его консультаций) или к преподавателю на практических занятиях. Не оставляйте «белых пятен» в освоении материала.

***Рекомендации по подготовке к практическим (семинарским) занятиям***

Проведение практических занятий осуществляется в компьютерных классах.

Студентам следует:

- приносить с собой на бумажных или электронных носителях рекомендованную преподавателем литературу к конкретному занятию;

- до очередного практического занятия по рекомендованным литературным источникам проработать теоретический материал, соответствующей темы занятия;

- при подготовке к практическим занятиям следует обязательно использовать не только лекции, но и другую учебную литературу;

- в начале занятий задать преподавателю вопросы по материалу, вызвавшему затруднения в его понимании и освоении, при решении задач, заданных для самостоятельного решения;

- в ходе семинара давать конкретные, четкие ответы по существу вопросов;

- на занятии доводить каждую задачу до окончательного решения, демонстрировать понимание проведенных расчетов (анализов, ситуаций), в случае затруднений обращаться к преподавателю.

***Методические рекомендации по выполнению различных форм самостоятельных домашних заданий***

Самостоятельная работа студентов включает в себя выполнение различного рода заданий, которые ориентированы на более глубокое усвоение материала изучаемой дисциплины. К выполнению заданий для самостоятельной работы предъявляются следующие требования: задания должны выполняться самостоятельно и представляться в установленный срок, а также соответствовать установленным требованиям по оформлению.

Студентам следует:

- руководствоваться графиком самостоятельной работы, определенным РПД;

- выполнять все плановые задания, выдаваемые преподавателем для самостоятельного выполнения, и разбирать на семинарах и консультациях неясные вопросы;

- использовать при подготовке нормативные документы Финансового университета;

- при подготовке к экзамену параллельно прорабатывать соответствующие теоретические и практические разделы дисциплины, фиксируя неясные моменты для их обсуждения на плановой консультации.

***Методические рекомендации по работе с литературой***

Любая форма самостоятельной работы студента (подготовка к семинарскому занятию, выполнение расчетно-аналитической работы, начинается с изучения соответствующей литературы как в библиотеке, так и дома.

К каждой теме учебной дисциплины подобрана основная и дополнительная литература.

При работе с литературой рекомендуется делать записи. Записи в той или иной форме не только способствуют пониманию и усвоению изучаемого материала, но и помогают вырабатывать навыки явного изложения в письменной форме тех или иных теоретических вопросов.

***Методические указания по проведению практических занятий***

По структуре практические занятия следует разделить на учебные и контрольные.

*Учебные практические занятия* структурно состоят из следующих компонент:

* проверка наличия выполненного задания самостоятельной работы каждого студента;
* выборочная проверка корректности выполнения домашнего задания;
* разбор типичных ошибок, возникших в самостоятельной работе;
* рассмотрение теоретических вопросов, связанных с текущим практическим занятием;
* разбор методов выполнения практических заданий и решения задач;
* корректировка заданий для самостоятельной работы студентов.

*Контрольные практические занятия* структурно состоят из следующих компонент:

* проведение аудиторных самостоятельных работ;
* подведение итогов и разбор типичных ошибок, возникших при выполнении самостоятельных работ.

Студенты должны обратить внимание на перечень основных контрольных мероприятий, которые проводятся в соответствии с рабочей программой дисциплины.

Конкретные сроки проведения этих мероприятий своевременно доводятся до сведения студентов.

# 11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

11.1. Комплект лицензионного программного обеспечения:

Пакет офисных программ (Astra Linux, Library Office и пр.);

Антивирус Kaspersky;

11.2. Современные профессиональные базы данных и информационные справочные системы:

Информационно-правовая система «Консультант Плюс»;

Информационно-правовая система «Гарант»;

Электронная энциклопедия: <http://ru.wikipedia.org/wiki/Wiki>

Система комплексного раскрытия информации «СКРИН»: [https://skrin.ru](https://skrin.ru/)

11.3. Сертифицированные программные и аппаратные средства защиты информации - не предусмотрены.

11.4. Язык программирования Python 3.x в среде Windows, библиотеки PyOpenCl, mpi4py.

11.5. Платформа для научных исследований, основанная на языке программирования Python, Anaconda.

11.6. Интерфейс передачи сообщений MS MPI <https://www.microsoft.com/en-us/download/details.aspx?id=100593>

11.7. Компилятор g++ <https://www.msys2.org>

11.8. Открытый стандарт для распараллеливания программ на языках Си, Си++ и Фортран OpenMP, реализация для компилятора g++ <https://packages.msys2.org/package/mingw-w64-x86_64-openmp?repo=mingw64>

# 12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Материально-техническая база Финансового университета, необходимая для осуществления образовательного процесса по данной дисциплине, в соответствии с требованиями ФОС ВО включает в себя специальные помещения для проведения лекций, семинарских занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации; помещения укомплектованы специализированной мебелью и техническими средствами обучения, необходимыми для представления информации большой аудитории.

Помещения для самостоятельной работы студентов включают в себя библиотеку с читальным залом, укомплектованную в соответствии с существующими нормами необходимой учебной и учебно-методической литературой и иными материалами; медиатеку с выходом в Интернет, компьютерные классы с возможностью работы в Интернет; аудитории для консультационной деятельности.