

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КАЛИНИНГРАДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»

А. В. Снытников

ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ И АЛГОРИТМИЗАЦИИ

Учебно-методическое пособие по изучению дисциплины для студентов
магистратуры по направлению подготовки
09.04.01 Информатика и вычислительная техника

Калининград
Издательство ФГБОУ ВО «КГТУ»
2023

Рецензент:

кандидат педагогических наук, доцент кафедры прикладной информатики ФГБОУ
ВО «Калининградский государственный технический университет»

Е. Ю. Заболотнова

Снытников, А. В.

Практикум по программированию и алгоритмизации: учебно-методическое пособие по изучению дисциплины для студентов магистратуры по направлению подготовки 09.04.01 Информатика и вычислительная техника / А. В. Снытников. – Калининград: Изд-во ФГБОУ ВО «КГТУ», 2023. – 13 с.

Учебно-методическое пособие является руководством по изучению дисциплины «Практикум по программированию и алгоритмизации» для студентов магистратуры по направлению подготовки 09.04.01 Информатика и вычислительная техника. Содержит характеристику дисциплины (цель и планируемые результаты изучения дисциплины, место дисциплины в структуре основной профессиональной образовательной программы, тематический план с описанием для каждой темы формы проведения занятия, вопросы для изучения, методические материалы к занятиям).

Учебно-методическое пособие по изучению дисциплины рекомендовано к использованию в учебном процессе в качестве локального электронного методического материала методической комиссией Института цифровых технологий 5 июля 2023 г., протокол № 8

© Федеральное государственное
бюджетное образовательное
учреждение высшего образования
«Калининградский государственный
технический университет», 2023 г.
© Снытников А. В., 2023 г.

ОГЛАВЛЕНИЕ

Введение	4
1. ТЕМАТИЧЕСКИЙ ПЛАН	5
2. СОДЕРЖАНИЕ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ	5
3. ТРЕБОВАНИЯ К АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ	10
Библиографический список.....	11
Приложение А.....	12
Контрольные вопросы по дисциплине	12

Введение

Дисциплина «Практикум по программированию и алгоритмизации» является частью образовательной программы магистратуры по направлению подготовки 09.04.01 Информатика и вычислительная техника и изучается студентами во втором семестре.

Целью освоения дисциплины «Практикум по программированию и алгоритмизации» является формирование у обучающихся знаний, умений и навыков в области программирования, в частности комбинирования программ, написанных на различных языках программирования, и для закрепления навыков использования данных языков на примере задач математического моделирования методом частиц в ячейках в упрощенной постановке.

В результате освоения дисциплины обучающийся должен:

знать:

- современный опыт использования технологий больших данных в части разработки новых методов, моделей, алгоритмов, технологий и инструментальных средств;

- синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования;

- технологии программирования;

- особенности выбранной среды программирования;

уметь:

- применять выбранные языки программирования для написания программного кода;

- использовать выбранную среду программирования и средства системы управления базами данных;

- использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода;

- осуществлять коммуникации с заинтересованными сторонами;

владеть:

- навыками создания программного кода в соответствии с техническим заданием (спецификацией);

- навыками оптимизации программного кода.

Дисциплина опирается на компетенции, полученные при изучении дисциплин «Программирование», «Программная инженерия», «Высокоуровневые технологии программирования».

Промежуточная аттестация по дисциплине проводится в форме зачета.

1. ТЕМАТИЧЕСКИЙ ПЛАН

Тематический план для студентов очной формы обучения приведен в таблице 1.

Таблица 1 – Трудоёмкость освоения дисциплины в третьем семестре по очной форме обучения

№ п/п	Раздел (модуль) дисциплины	Контактная работа с преподавателем					СРС	Подготовка и аттестация в период сессии	
		ЛК	ЛР	ПР	РЭ	КА			
1	Основы языка программирования C/C++, часть 1		10						
2	Основы языка программирования C/C++, часть 2		10						
3	Метод частиц в ячейках. Введение		10						
4	Метод частиц в ячейках. Основные элементы		10						
5	Анализ производительности приложения		10						
6	Вычисление производительности ЭВМ		10						
ИТОГО:			60	4	0,15		79,85		
Всего		144							

2. СОДЕРЖАНИЕ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ

Курс предполагает выполнение лабораторных работ, темы которых приведены ниже.

Тема 1. Основы языка программирования C/C++, часть 1

Ключевые вопросы темы

1. Элементы языка.

2. Выражения.
3. Операторы.
4. Указатели.
5. Операции для работы с динамической памятью.
6. Объявления и определения.
7. Область существования имени.
8. Область видимости.
9. Классы памяти.

Источники

[1, с. 1–106].

Методические рекомендации

В начале изучения дисциплины «Практикум по программированию и алгоритмизации» необходимо повторить курс по языку программирования C++, Или провести краткий вводный курс по этому языку тем студентам, чьи знания недостаточны для освоения дальнейшего материала.

Вопросы для самоконтроля

1. Какие существуют основные элементы языка C++?
2. Назовите основные типы выражений в C++.
3. Какие действия производят операторы ++, +=, >>, &, унарный *?
4. В чем отличие указателей от массивов?
5. Для чего нужна динамическая память?
6. Чем отличаются объявления и определения переменной?

Тема 2. Основы языка программирования C/C++, часть 2

Ключевые вопросы темы

1. Функции.
2. Классы.
3. Производные классы.
4. Полиморфизм.
5. Классы и шаблоны.

Источники

[1, с. 122–201].

Методические рекомендации

В рамках данной темы студентам необходимо изучить, что такое классы, с дальнейшей добавкой шаблонов классов, наследования и полиморфизма. Вместе с тем следует избегать чрезмерного погружения в объектно-ориентированное программирование.

Вопросы для самоконтроля

1. Как оформляется функция в языке C++?
2. Для чего нужны классы?
3. Какие существуют разновидности наследования в C++?
4. Что такое полиморфизм?
5. Какую задачу решают шаблоны?

Тема 3. Метод частиц в ячейках. Введение

Ключевые вопросы темы

1. Определяющие уравнения.
2. Модельные частицы и их свойства.
3. Общая схема.
4. Прямая и обратная интерполяция.
5. Погрешность вычислений.

Источники

[2, с. 28–51].

Методические рекомендации

Перед началом изучения данной темы необходимо ознакомиться с основными уравнениями, определяющими ход вычисления по методу частиц, а также с предназначением данного метода и областью его применимости.

Затем можно рассмотреть общую схему вычислений по методу частиц, его основные этапы, как выбирается количество частиц и количество узлов сетки. И после этого можно переходить к рассмотрению отдельных этапов расчета по методу частиц в ячейках.

При изучении данной темы следует избегать погружения в многочисленные подробности метода частиц в ячейках, и наоборот, стремиться к тому, чтобы студенты могли разработать собственную корректную реализацию метода частиц в ячейках и максимально ее осмыслить.

Вопросы для самоконтроля

1. Какой смысл имеет уравнение Власова?
2. Как выбирается количество модельных частиц?
3. Назовите основные этапы метода частиц в ячейках.
4. Какой смысл имеет отдельная модельная частица?
5. В чем причина возникновения нефизических эффектов?

Тема 4. Метод частиц в ячейках. Основные элементы

Ключевые вопросы темы

1. Расчет электрического поля.
2. Расчет плотности массы и заряда
3. Интегрирование уравнений движения модельных частиц.

Источники

[2, с. 190–211].

Методические рекомендации

Начинать изучение данной темы целесообразно с рассмотрения трех способов расчета электрического поля – численного решения уравнения Пуассона с помощью обращения матрицы СЛАУ, преобразования Фурье и прямого расчета силы по закону Кулона.

Затем следует рассмотреть методику определения плотности массы и заряда, рассмотрев при этом эффект от применения ядер высокого порядка и обращая особое внимание на возможности автоматизированного ускорения этого этапа.

Далее в ходе изучения данной темы следует продолжить рассмотрением вариантов интегрирования уравнений движения модельных частиц, обращая особое внимание на интенсивность обращений к оперативной памяти.

Вопросы для самоконтроля

1. В чем преимущество метода расчета электрического поля с помощью решения уравнения Пуассона?
2. Какие существуют ядра модельных частиц?
3. В чем преимущества методов высокого порядка для интегрирования уравнений движения модельных частиц?
4. Как определяется величина погрешности метода частиц в ячейках исходя из количества модельных частиц?

Тема 5. Анализ производительности приложения

Ключевые вопросы темы

1. Измерение времени средствами приложения.
2. Профилировка.
3. Инструменты GNU Profiler, Valgrind.
4. Intel VTune.
5. Профилировщики для языка Python.

Методические рекомендации

Целью изучения данной темы является научить студентов самостоятельно измерять вычислительную производительность своего приложения. Для этого вначале рассматриваются средства для измерения времени внутри программы, затем – несколько профилировщиков, как свободно распространяемых, так и коммерческих. После изучения этих вопросов следует добиться того, чтобы результаты измерения производительности вручную показывали результаты, сравнимые с результатами профилировки.

Вопросы для самоконтроля

1. Какие существуют функции языка C для измерения времени?
2. Что такое профилировка?
3. В каких единицах показывает время профилировщик GNU Profiler?
4. Может ли Intel VTune работать с многопоточными и параллельными программами?
5. Можно ли использовать профилировщики для Python в среде Google Colab?

Тема 6. Вычисление производительности ЭВМ

Ключевые вопросы темы

1. Вычисление производительности процессора.
2. Вычисление производительности системы памяти.
3. Вычисление производительности коммуникационной системы.
4. Сравнение с известными тестами производительности: Linpack, HPCG и др.

Источники

[4].

Методические рекомендации

Целью рассмотрения данной темы является изучение методики комплексного тестирования производительности ЭВМ, в частности сервера, на основе приложения, реализующего метод частиц в ячейках. Рассмотрение данной темы необходимо начинать с определения количества арифметических операций и количества байт, записываемых в память в расчете на каждую модельную частицу в программе, реализующей метод частиц в ячейках, созданной студентом в предшествующей части курса.

Далее на основе измерения производительности программы, реализующей метод частиц в ячейках, с учетом известной сложности каждого этапа, определяется производительность ЭВМ, на которой выполняется данная программа. Важно научиться находить положение мгновенного центра скоростей (МЦС) в любой момент времени движения тела.

Рекомендуется рассмотреть частные случаи нахождения мгновенного центра скоростей. Рассмотреть несколько примеров нахождения МЦС.

Необходимо также ознакомиться с теоремой о проекциях скоростей двух точек тела.

Затем необходимо рассмотреть, что называется мгновенным центром ускорений. Далее можно переходить к определению ускорений. И завершить изучение данной темы рассмотрением примера решения задачи по определению ускорения точки тела, совершающего плоскопараллельное движение.

Вопросы для самоконтроля

1. Как можно измерить количество арифметических операций, выполняемых в программе?
2. Какие существуют способы для измерения производительности системы памяти?
3. Какие существуют способы для измерения производительности процессора?
4. Какие существуют способы для измерения производительности коммуникационной системы?
5. Каковы достоинства и недостатки теста LinPack?

3. ТРЕБОВАНИЯ К АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

Текущий контроль предназначен для проверки хода и качества усвоения студентами учебного материала и стимулирования их учебной работы. Он может

осуществляться в ходе всех видов занятий в форме, избранной преподавателем или предусмотренной рабочей программой дисциплины.

Текущий контроль предполагает постоянный контроль преподавателем качества усвоения учебного материала, активизацию учебной деятельности студентов на занятиях, побуждение их к самостоятельной систематической работе. Он необходим обучающимся для самоконтроля на разных этапах обучения.

К оценочным средствам для промежуточной аттестации, проводимой в форме зачета во втором семестре, относятся вопросы, которые при необходимости могут быть использованы для защиты лабораторных работ и проведения зачета.

Библиографический список

1. Андрианова, А. А. Алгоритмизация и программирование. Практикум: учебное пособие / А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. – Санкт-Петербург: Лань, 2022. – 240 с. – Режим доступа: для авториз. пользователей. – Лань: электронно-библиотечная система. – URL: <https://e.lanbook.com/book/206258> (дата обращения: 23.02.2023). – ISBN 978-5-8114-3336-0. – Текст: электронный.
2. Полупанов, Д. В. Программирование в Python 3: учебное пособие / Д. В. Полупанов, С. Р. Абдюшева, А. М. Ефимов. – Уфа: БашГУ, 2020. – 164 с. – Режим доступа: для авториз. пользователей. – Лань: электронно-библиотечная система. – URL: <https://e.lanbook.com/book/179915> (дата обращения: 23.02.2023). – ISBN 978-5-7477-5230-6. – Текст: электронный.
3. Царёв, Р. Ю. Алгоритмы и структуры данных (СДИО): учебник / Р. Ю. Царёв, А. В. Прокопенко; Сибирский федеральный университет. – Красноярск: Сибирский федеральный университет (СФУ), 2016. – 204 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=497016> (дата обращения: 23.02.2023). – ISBN 978-5-7638-3388-1. – Текст: электронный.
4. Хахаев, И. А. Практикум по алгоритмизации и программированию на Python: курс: учебное пособие / И. А. Хахаев. – 2-е изд., исправ. – Москва: Национальный Открытый Университет «ИНТУИТ», 2016. – 179 с. – Режим доступа: по подписке. <https://biblioclub.ru/index.php?page=book&id=429256> (дата обращения: 23.02.2023). – Библиогр. в кн. – Текст: электронный.

Контрольные вопросы по дисциплине

1. Какие существуют основные элементы языка C++?
2. Назовите основные типы выражений в C++.
3. Какие действия производят операторы ++, +=, >>, &, унарный *?
4. В чем отличие указателей от массивов?
5. Для чего нужна динамическая память?
6. Чем различаются объявления и определения переменной?
7. Как оформляется функция в языке C++?
8. Для чего нужны классы?
9. Какие существуют разновидности наследования в C++?
10. Что такое полиморфизм?
11. Какую задачу решают шаблоны?
12. Какой смысл имеет уравнение Власова?
13. Как выбирается количество модельных частиц?
14. Назовите основные этапы метода частиц в ячейках.
15. Какой смысл имеет отдельная модельная частица?
16. В чем причина возникновения нефизических эффектов?
17. В чем преимущество метода расчета электрического поля с помощью решения уравнения Пуассона?
18. Какие существуют ядра модельных частиц?
19. В чем преимущества методов высокого порядка для интегрирования уравнений движения модельных частиц?
20. Как определяется величина погрешности метода частиц в ячейках исходя из количества модельных частиц?
21. Какие существуют функции языка C для измерения времени?
22. Что такое профилировка?
23. В каких единицах показывает время профилировщик GNU Profiler?
24. Может ли Intel VTune работать с многопоточными и параллельными программами?
25. Можно ли использовать профилировщики для Python в среде Google Colab?
26. Как можно измерить количество арифметических операций, выполняемых в программе?
27. Какие существуют способы для измерения производительности системы памяти?
28. Какие существуют способы для измерения производительности процессора?
29. Какие существуют способы для измерения производительности коммуникационной системы?
30. Каковы достоинства и недостатки теста LinPack?

Локальный электронный методический материал

Алексей Владимирович Снытников

ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ И АЛГОРИТМИЗАЦИИ

Редактор М. А. Дмитриева

Уч.-изд. л. 0,6. Печ. л. 0,8.

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Калининградский государственный технический университет».
236022, Калининград, Советский проспект, 1