

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
кибербезопасности информационных систем



Кенин С. Л.
22.03.2024 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1.О.56.03 Разработка прикладного программного обеспечения
для компьютерных систем

1. Код и наименование направления подготовки/специальности:
10.05.01 Компьютерная безопасность
2. Профиль подготовки / специализация / магистерская программа:
N 4 «Безопасность компьютерных систем и сетей»
(по отрасли или в сфере профессиональной деятельности)
3. Квалификация (степень) выпускника: специалист
4. Форма обучения: очная
5. Кафедра, отвечающая за реализацию дисциплины: кибербезопасности информационных систем
6. Составители программы: Каплиева Наталья Алексеевна
кандидат физико-математических наук, доцент
7. Рекомендована: НМС факультета ПММ, протокол № 5 от 22.03.2024

отметки о продлении вносятся вручную)

8. Учебный год: 2026/2027

Семестр(ы): 6

9. Цели и задачи учебной дисциплины:

Цель – изучение методов разработки прикладного программного обеспечения и приобретение навыков применения средств и методов контроля разработки программного обеспечения с использованием технологий параллельного программирования.

Задачи:

- Изучение классов ошибок, допускаемых при разработке прикладного программного обеспечения.
- Изучение методов устранения ошибок и уязвимостей в программном обеспечении.
- Разработка оптимизированных алгоритмов прикладного программного обеспечения с использованием технологий параллельного программирования.
- Изучение критериев и показателей оценки процессов разработки прикладного программного обеспечения.
- Изучение методов контроля разработки прикладного программного обеспечения.
- Разработка организационно-распорядительной и эксплуатационной документации, используемой в процессе разработки прикладного программного обеспечения.

10. Место учебной дисциплины в структуре ООП: Дисциплина относится к обязательной части блока Б1 (дисциплина специализации) программы специалитета. Изучение данного курса должно базироваться на знаниях обучающимися материала курса «Информатика», «Введение в программирование». Дисциплина является базовой для изучения курса «Разработка безопасного программного обеспечения».

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ОПК-4.1.	Способен организовывать защиту информации в компьютерных системах и сетях (по областям применения)	ОПК-4.1.3	Способен использовать языки и системы программирования, инструментальные средства при обеспечении защиты информации в компьютерных системах при решении различных профессиональных, исследовательских и прикладных задач	Знать: – <i>классы ошибок, допускаемых при разработке прикладного программного обеспечения;</i> – <i>методы устранения ошибок и уязвимостей в программном обеспечении.</i> Уметь: – <i>разрабатывать оптимизированные алгоритмы прикладного программного обеспечения с использованием технологий параллельного программирования;</i>
ОПК-4.3	Способен разрабатывать и анализировать корректность политики информационной безопасности компьютерных систем и сетей (по областям применения)	ОПК-4.3.5	Способен применять программные средства прикладного, системного и специального назначения при разработке и анализе политики информационной безопасности	– <i>разрабатывать организационно-распорядительную и эксплуатационную документацию, используемую в процессе разработки прикладного программного обеспечения.</i> Владеть: – <i>методами контроля разработки прикладного программного обеспечения.</i>

12. Объем дисциплины в зачетных единицах/час.(в соответствии с учебным планом) — 3/108.

Форма промежуточной аттестации(зачет/экзамен) зачет с оценкой.

13. Трудоемкость по видам учебной работы

Вид учебной работы		Трудоемкость	
		Всего	По семестрам
			6 семестр
Аудиторные занятия		72	72
в том числе:	лекции	36	36
	практические	0	0
	лабораторные	36	36
Самостоятельная работа		36	36
в том числе: курсовая работа (проект)			
Форма промежуточной аттестации (зачет – 0 час. / экзамен – 0 час.)		0	0
Итого:		108	108

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК *	
1. Лекции				
1.1	Общая характеристика параллельных вычислительных систем.	Классификация вычислительных систем. Процессы и потоки.	Онлайн-курс «Разработка прикладного программного обеспечения для компьютерных систем_КБИС» https://edu.vsu.ru/course/view.php?id=10369	
1.2	Создание потоков.	Создание потоков средствами операционной системы. Создание потоков средствами языка программирования		
1.3	Проектирование параллельных взаимодействующих процессов.	Атомарный доступ: семейство Interlocked-функций. Синхронизация потоков с использованием объектов ядра.		
1.4	Технология OpenMP	Создание взаимодействующих потоков средствами технологии OpenMP		
3. Лабораторные работы				
3.1	Создание потоков.	Создание потоков средствами операционной системы.		
3.2	Создание потоков.	Создание потоков средствами языка программирования		
3.3	Проектирование параллельных взаимодействующих процессов.	Атомарный доступ: семейство Interlocked-функций.		
3.4	Проектирование параллельных взаимодействующих процессов.	Синхронизация потоков с использованием объектов ядра. Разработка пула потоков		
3.5	Классические задачи параллельного программирования	Производители-Потребители		
3.6	Технология OpenMP	Создание взаимодействующих потоков средствами технологии OpenMP		

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование раздела дисциплины	Виды занятий (часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1	Общая характеристика параллельных вычисли-	2		2	2	6

	тельных систем.					
2	Создание потоков.	12		12	12	36
3	Проектирование параллельных взаимодействующих процессов.	16		16	16	48
4	Технология OpenMP	6		6	6	18
Итого:		36		36	36	108

14. Методические указания для обучающихся по освоению дисциплины

Работа с конспектами лекций, выполнение лабораторных заданий, заданий текущей и промежуточной аттестаций.

При использовании дистанционных образовательных технологий и электронного обучения выполнять все указания преподавателей по работе на LMS-платформе, своевременно подключаться к online-занятиям, соблюдать рекомендации по организации самостоятельной работы

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины (список литературы оформляется в соответствии с требованиями ГОСТ и используется общая сквозная нумерация для всех видов источников)

а) основная литература:

№ п/п	Источник
1	Туральчук, К.А. Параллельное программирование с помощью языка C# / К.А. Туральчук. - 2-е изд., испр. – М. : Национальный Открытый Университет «ИНТУ-ИТ», 2016. - 190 с. : – URL: https://biblioclub.lib.vsu.ru/index.php?page=book&id=429098 (16.09.2016).
2	Алексеев, А.А. Основы параллельного программирования с использованием Visual Studio 2010 / А.А. Алексеев. - 2-е изд., испр. – М. : Национальный Открытый Университет «ИНТУИТ», 2016. – 332 с. – URL: https://biblioclub.lib.vsu.ru/index.php?page=book&id=428829 (16.09.2016).
3	Биллиг, В.А. Параллельные вычисления и многопоточное программирование / В.А. Биллиг. - 2-е изд., испр. – М. : Национальный Открытый Университет «ИНТУИТ», 2016. – 311 с. – URL: https://biblioclub.lib.vsu.ru/index.php?page=book&id=428948 (16.09.2016).
4	Левин, М.П. Параллельное программирование с использованием OpenMP : учебное пособие / М.П. Левин. – М. : Интернет-Университет Информационных Технологий, 2008. - 120 с. – URL: https://biblioclub.lib.vsu.ru/index.php?page=book&id=233111 (16.09.2016).
5	Антонов, А.С. Параллельное программирование с использованием технологии MPI : курс / А.С. Антонов. – М. : Интернет-Университет Информационных Технологий, 2008. - 71 с. – URL: https://biblioclub.lib.vsu.ru/index.php?page=book&id=233577 (16.09.2016).

б) дополнительная литература:

№ п/п	Источник
6	Хьюз К. Параллельное и распределенное программирование с использованием C++ / К. Хьюз, Т. Хьюз. – М. : Вильямс, 2004. – 672 с.
7	Параллельное программирование в MPI / В.Д. Корнеев. – М. ; Ижевск : Ин-т компьютерных исследований, 2003. – 303 с.

в) информационные электронно-образовательные ресурсы:

№ п/п	Источник
8	www.lib.vsu.ru – ЗНБ ВГУ
9	Богачёв К.Ю. Основы параллельного программирования: учебное пособие / К.Ю. Богачёв – М. : "Лаборатория знаний", 2013. – 342 с. Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=42626
10	Федотов И.Е. Модели параллельного программирования. – М. : СОЛОН-Пресс, 2012. – 384 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=13807

* Вначале указываются ЭБС, с которыми имеются договора у ВГУ, затем открытые электронно-образовательные ресурсы

16. Перечень учебно-методического обеспечения для самостоятельной работы (учебно-методические рекомендации, пособия, задачки, методические указания по выполнению лабораторных и контрольных работ)

№ п/п	Источник
1	<i>Абрамов, Г. В. Параллельное программирование : лабораторный практикум / Г. В. Абрамов, Н. А. Каплиева. — Воронеж : Издательский дом ВГУ, 2021. — 75 с.</i>

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

При реализации дисциплины используются модульно-рейтинговая и личностно-ориентированные технологии обучения (ориентированные на индивидуальность студента, компьютерные и коммуникационные технологии). В рамках дисциплины предусмотрены следующие виды лекций: информационная, лекция-визуализация, лекция с применением обратной связи.

Дисциплина реализуется с применением электронного обучения и дистанционных образовательных технологий, для организации самостоятельной работы обучающихся используется онлайн-курс, размещенный на платформе Электронного университета ВГУ (LMS moodle), а также другие Интернет-ресурсы, приведенные в п.15в

18. Материально-техническое обеспечение дисциплины:

Лекционная аудитория должна быть оборудована учебной мебелью, компьютером, мультимедийным оборудованием (проектор, экран, средства звуковоспроизведения), допускается переносное оборудование.

Лабораторные занятия должны проводиться в специализированной аудитории, оснащенной учебной мебелью и персональными компьютерами с доступом в сеть Интернет (компьютерные классы, студии), мультимедийным оборудованием (мультимедийный проектор, экран, средства звуковоспроизведения). Число рабочих мест в аудитории должно быть таким, чтобы обеспечивалась индивидуальная работа студента на отдельном персональном компьютере.

Для самостоятельной работы необходимы компьютерные классы, помещения, оснащенные компьютерами с доступом к сети Интернет.

Программное обеспечение (см. файл МТО):

- ОС Windows 8 (10)
- Интернет-браузер (Google Chrome, Mozilla Firefox)
- Microsoft Visual Studio Community Edition (свободное и/или бесплатное ПО)
- Adobe Reader (свободное и/или бесплатное ПО)

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

- контрольная работа
- тесты.

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.	Общая характеристика па-	ОПК-4.3	ОПК-4.3.5	КИМы для про-

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
	параллельных вычислительных систем.			ведения текущей аттестации Задания для лабораторных работ
2.	Создание потоков.	ОПК-4.3	ОПК-4.3.5	
3.	Проектирование параллельных взаимодействующих процессов.	ОПК-4.1	ОПК-4.1.3	
4.	Технология OpenMP	ОПК-4.1	ОПК-4.1.3	
Промежуточная аттестация форма контроля - зачет с оценкой				КИМы для проведения итоговой аттестации

* В графе «ФОС» в обязательном порядке перечисляются оценочные средства текущей и промежуточной аттестаций.

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

Оценка знаний, умений и навыков, характеризующая этапы формирования компетенций в рамках изучения дисциплины осуществляется в ходе текущей и промежуточной аттестаций.

Текущая аттестация проводится в соответствии с Положением о текущей аттестации обучающихся по программам высшего образования Воронежского государственного университета. Текущая аттестация проводится в формах: устного опроса; защиты лабораторных работ, выполнения контрольных работ.

Промежуточная аттестация проводится в соответствии с Положением о промежуточной аттестации обучающихся по программам высшего образования. Промежуточная аттестация по итогам освоения дисциплины проводится в форме зачета с оценкой и экзамена. Для получения положительной итоговой оценки необходимо выполнение всех лабораторных и контрольных работ.

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью лабораторных и контрольных работ.

Текущая аттестация проводится на занятии одновременно во всей учебной группе в виде теста в электронной образовательной среде «Электронный университет ВГУ», адрес курса — <https://edu.vsu.ru/course/view.php?id=10369>, адрес теста текущей аттестации — <https://edu.vsu.ru/mod/assign/view.php?id=1186371> Тест составляется из материалов ФОСа, формируется системой автоматически путём добавления случайных вопросов, количество которых соответствует имеющимся образцам билетов. Большая часть вопросов проверяется автоматически, проверки преподавателем с ручным оцениванием требуют только отдельные вопросы, представленные в форме эссе. Ограничение по времени на каждую попытку — 1 час 30 минут

Примеры контрольных работ

Примеры контрольных работ

Тема «Параллелизм в вычислительных системах»

Вариант 1

Задание 1

Опишите классификацию вычислительных систем (систематика Флинна).

Задание 2

Опишите алгоритм использования объекта синхронизации «Мьютекс».

Вариант 2

Задание 1

Опишите классификацию многопроцессорных систем, основанную на используемых способах организации оперативной памяти в этих системах

Задание 2

Опишите алгоритм использования объекта синхронизации «Событие».

Примеры лабораторных работ

1. Разработайте и реализуйте параллельный алгоритм для вычисления определенного интеграла методом трапеций.

Вариант 1. Поток создается с помощью WinAPI функций.

Вариант 1. Поток создается с помощью специализированных классов языка программирования.

2. Разработайте и реализуйте алгоритм синхронизированного доступа к стеку, используя объект синхронизации «Семафор».

Критерии оценивания компетенций	Уровень сформированности компетенций	Шкала оценок
Сформированные знания классов ошибок, допускаемых при разработке прикладного программного обеспечения; методов устранения ошибок и уязвимостей в программном обеспечении. Сформированные умения разрабатывать оптимизированных алгоритмов прикладного программного обеспечения с использованием технологий параллельного программирования. Сформированные навыки контроля разработки прикладного программного обеспечения.	<i>Повышенный уровень</i>	<i>Отлично</i>
Сформированные, но содержащие отдельные пробелы, знания классов ошибок, допускаемых при разработке прикладного программного обеспечения; методов устранения ошибок и уязвимостей в программном обеспечении. Сформированные, но содержащие отдельные пробелы, умения разрабатывать оптимизированных алгоритмов прикладного программного обеспечения с использованием технологий параллельного программирования. Сформированные, но содержащие отдельные пробелы, навыки контроля разработки прикладного программного обеспечения.	<i>Базовый уровень</i>	<i>Хорошо</i>
Неполное представление о классах ошибок, допускаемых при разработке прикладного программного обеспечения; методах устранения ошибок и уязвимостей в программном обеспечении. Успешное, но не системное умение разрабатывать оптимизированных алгоритмов прикладного программного обеспечения с использованием технологий параллельного программирования. Неполное владение навыками контроля разработки прикладного программного обеспечения.	<i>Пороговый уровень</i>	<i>Удовлетворительно</i>
Фрагментарные знания или отсутствие знаний. Фрагментарные умения или отсутствие умений. Фрагментарные навыки или отсутствие навыков	–	<i>Неудовлетворительно</i>

20.3 Фонд оценочных средств сформированности компетенций студентов, рекомендуемый для проведения диагностических работ

ОПК-4.1. Способен организовывать защиту информации в компьютерных системах и сетях (по областям применения).

ОПК-4.3. Способен разрабатывать и анализировать корректность политики информационной безопасности компьютерных систем и сетей (по областям применения).

Вопросы с вариантами ответов (закрытые)

1. Синхронизация потоков заключается:

А. В согласованном выполнении системных вызовов этими потоками

- B. В согласовании их скоростей путем приостановки потоков
- C. В согласованном доступе к аппаратным средствам

Ответ: B

2. До прерывания выполнения потока ОС запоминает его:
- A. Коды
 - B. Описатель
 - C. Контекст
 - D. Образ
 - E. Данные

Ответ: C

3. При создании потока ОС сразу переводит его в состояние:
- A. Готовности
 - B. Выполнения
 - C. Ожидания

Ответ: A

4. В критические секции:
- A. потоки входят последовательно в произвольном порядке
 - B. потоки входят последовательно в соответствии с порядковым номером потока
 - C. все потоки входят одновременно

Ответ: A

5. Термин race condition (условие гонки) описывает ситуацию, когда:
- A. несколько потоков работают с разделяемыми данными, и конечный результат зависит от соотношения скоростей потоков
 - B. несколько потоков работают с данными, локализованными для каждого из потоков, и конечный результат собирается в одну переменную
 - C. не хватает памяти для размещения данных

Ответ: A

6. Термин deadlock (тупик) описывает ситуацию, когда:
- A. все потоки завершили свою работу
 - B. возникла взаимная блокировка потоков, ожидающих наступления некоторого события для продолжения работы
 - C. недостаточно ресурсов для запуска всех потоков

Ответ: B

7. Какая директива OpenMP может быть использована для синхронизации потоков:
- A. `#pragma omp barrier`
 - B. `#pragma omp waitall`
 - C. `#pragma omp wait`

Ответ: A

8. Может ли в OpenMP программе случиться ситуация, когда один поток закончит выполнение двух параллельных регионов, в то время как другой поток все еще выполняет первый параллельный регион, при условии что все потоки запущены средствами OpenMP?

- A. нет, после окончания каждого параллельного региона все потоки должны закончить его выполнение
- B. нет, в программе может быть только один параллельный регион
- C. да, каждый поток работает независимо

Ответ: A

9. Может ли в OpenMP программе случиться ситуация, когда один поток закончит выполнение двух параллельных циклов, в то время как другой поток все еще выполняет первый параллельный цикл, при условии что все потоки запущены средствами OpenMP?

- A. нет, после окончания каждого параллельного цикла все потоки должны закончить его выполнение
- B. нет, в программе может быть только один параллельный цикл
- C. да, каждый поток работает независимо
- D. да, но только если первый цикл объявлен с условием `nowait` и оба цикла принадлежат одному параллельному региону

Ответ: D

10. Можно ли в OpenMP параллельном цикле выделить часть кода, которую потоки будут выполнять в определенном порядке, в то время как остальной код цикла будет выполняться параллельно?

- A. нет, в параллельном цикле код выполняется потоками в неопределенном порядке
- B. нет, можно только выделять части кода для последовательного выполнения с помощью критических секций, но неопределенно, в каком порядке потоки будут выполнять выделенный код
- C. да, одну из секций кода можно выделить для последовательного выполнения потоками в порядке следования итераций с помощью директивы `ordered`

Ответ: C

Вопросы с кратким текстовым ответом (открытые)

1. Процедура определения в какой момент необходимо прервать выполнение текущего активного потока и какому потоку предоставить возможность выполняться

Ответ: Планирование

2. Процедура переключения процессора с одного потока на другой

Ответ: Диспетчеризация

3. Поток, сделавший синхронный (блокирующий) вызов, переводится планировщиком ОС в состояние ожидания, а после завершения обработки вызова – в состояние

Ответ: Готовности

4. На какой параметр следует заменить параметр `private`, чтобы локальные копии переменной `x` в каждом потоке имели то же значение, что и исходная переменная `x`?

Ответ: `firstprivate`

5. Какое условие следует использовать в OpenMP в ситуации, когда необходимо, чтобы основной поток не ждал завершения остальных потоков

Ответ: `nowait`

6. Какую директиву в OpenMP следует использовать в параллельных циклах, чтобы указать, что блок должен исполняться в строго фиксированной последовательности

Ответ: ordered

Критерии и шкалы оценивания заданий ФОС:

Для оценивания выполнения заданий используется балльная шкала:

1) закрытые задания (тестовые с вариантами ответов, средний уровень сложности):

- 1 балл – указан верный ответ;
- 0 баллов – указан неверный ответ (полностью или частично неверный).

2) открытые задания (тестовые с кратким текстовым ответом, повышенный уровень сложности):

- 2 балла – указан верный ответ;
- 0 баллов – указан неверный ответ (полностью или частично неверный).

Задания раздела 20.3 рекомендуются к использованию при проведении диагностических работ с целью оценки остаточных результатов освоения данной дисциплины (знаний, умений, навыков).