

Министерство науки и высшего образования Российской Федерации  
 Федеральное государственное бюджетное образовательное  
 учреждение высшего образования  
 «Комсомольский-на-Амуре государственный университет»

УТВЕРЖДАЮ

Декан факультета  
 факультета компьютерных технологий  
 (наименование факультета)

Я.Ю. Григорьев

(подпись, ФИО)

« 19 » 05 2021 г.

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

### «Параллельное программирование»

Направление подготовки	09.03.01 Информатика и вычислительная техника	
Направленность (профиль) образовательной программы	Программное обеспечение средств вычислительной техники и автоматизированных систем	
Квалификация выпускника	Бакалавр	
Год начала подготовки (по учебному плану)	2021	
Форма обучения	Заочная форма	
Технология обучения	Традиционная	
Курс	Семестр	Трудоемкость, з.е.
3,4	6,7	4
Вид промежуточной аттестации	Обеспечивающее подразделение	
Зачет с оценкой	Кафедра «Математическое обеспечение и применение ЭВМ»_(Архив)	

Комсомольск-на-Амуре 2021

Разработчик рабочей программы:

Кандидат технических наук



(подпись)

Трещев  
И.А

СОГЛАСОВАНО:

Заведующий кафедрой

Кафедра «Проектирование, управление и разработка  
информационных систем»



(подпись)

А.Н.  
Петрова

## 1 Введение

Рабочая программа и фонд оценочных средств дисциплины «Параллельное программирование» составлены в соответствии с требованиями федерального государственного образовательного стандарта, утвержденного приказом Минобрнауки Российской Федерации 19.09.2017 № 929, и основной профессиональной образовательной программы подготовки «Программное обеспечение средств вычислительной техники и автоматизированных систем» по направлению подготовки «09.03.01 Информатика и вычислительная техника».

Практическая подготовка осуществляется на основе профессионального стандарта 06.001 «Программист». Обобщенная трудовая функция: D - Разработка требований и проектирование программного обеспечения Профессиональный стандарт

Задачи дисциплины	<ol style="list-style-type: none"> <li>1. Получение практических навыков создания параллельных процессов и работы с ними.</li> <li>2. Получение практических навыков программной реализации многопоточных приложений.</li> <li>3. Получение практических навыков программной реализации методов синхронизации параллельных процессов и потоков.</li> <li>4. Получение практических навыков построения и использования сетей Петри.</li> </ol>
Основные разделы / темы дисциплины	<p><b>Многопроцессорные вычислительные системы:</b> Системы параллельной обработки данных и архитектура процессоров, Параллельное выполнение процессов, Программирование параллельных вычислительных процессов,</p> <p><b>Разработка многопоточных приложений:</b> Загрузка и выполнение параллельных потоков , Разработка многопоточных программ , Методы распараллеливания рекурсивных подпрограмм ,</p> <p><b>Классические задачи синхронизации:</b> Семафоры и события, Решения проблемы сериализации с помощью семафоров,</p> <p><b>Моделирование параллельных вычислений:</b> Сети Петри и их применение. Волновые системы, Построение и программирование волновых систем с помощью сетей Петри, , Расчетно-графическая работа</p> <p><b>Промежуточная аттестация:</b></p>

## 2 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Процесс изучения дисциплины «Параллельное программирование» направлен на формирование следующих компетенций в соответствии с ФГОС ВО и основной образовательной программой (таблица 1):

Таблица 1 – Компетенции и индикаторы их достижения

Код и наименование компетенции	Индикаторы достижения	Планируемые результаты обучения по дисциплине
--------------------------------	-----------------------	---

Общепрофессиональные		
ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.	ОПК-9.1 Знает методики использования программных средств для решения практических задач ОПК-9.2 Умеет использовать программные средства для решения практических задач ОПК-9.3 Владеет навыками использования программных средств для решения практических задач	Знает основные архитектуры параллельных систем, методики использования программных средств для решения практических задач с использованием параллельных алгоритмов Умеет использовать программные средства для создания программ использующих параллелизм Владеет навыками программирования для многопроцессорных систем
Профессиональные		

### 3 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Параллельное программирование» изучается на 4 курсе, 7 семестре.

Дисциплина входит в состав блока 1 «Дисциплины (модули)» и относится к базовой части.

Для освоения дисциплины необходимы знания, умения, навыки и / или опыт практической деятельности, сформированные в процессе изучения дисциплин / практик: «Логическое программирование», «Языки программирования», «Инструменты подготовки ресурсов приложений», «Технологии и методы программирования», «Операционные системы», «Базы данных», «Интернет-технологии», «Сети и телекоммуникации», «Тестирование программного обеспечения», «Объектно-ориентированное программирование», «Технология разработки программного обеспечения», «Учебная практика (ознакомительная практика)».

Знания, умения и навыки, сформированные при изучении дисциплины «Параллельное программирование», будут востребованы при изучении последующих дисциплин: «Компьютерная графика», «Преддипломная практика», «Специальные средства современных систем управления базами данных», «Разработка интерфейса пользователя», «Альтернативные операционные системы», «Производственная практика (технологическая (проектно-технологическая) практика)».

Дисциплина «Параллельное программирование» в рамках воспитательной работы направлена на формирование у обучающихся активной гражданской позиции, уважения к правам и свободам человека, знания правовых основ и законов, воспитание чувства ответственности или умения аргументировать, самостоятельно мыслить, развивает творчество, профессиональные умения или творчески развитой личности, системы осознанных знаний, ответственности за выполнение учебно-производственных заданий и т.д.

**4 Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся** Общая трудоемкость (объем) дисциплины составляет 4 з.е., 144 акад. час.

Распределение объема дисциплины (модуля) по видам учебных занятий представлено в таблице 2.

Таблица 2 – Объем дисциплины (модуля) по видам учебных занятий

Объем дисциплины	Всего академических часов
Общая трудоемкость дисциплины	144
<b>Контактная аудиторная работа обучающихся с преподавателем (по видам учебных занятий), всего</b>	10
<b>В том числе:</b>	
<b>занятия лекционного типа (лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации педагогическими работниками)</b>	4

занятия семинарского типа (семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия)	6
Самостоятельная работа обучающихся и контактная работа, включающая групповые консультации, индивидуальную работу обучающихся с преподавателями (в том числе индивидуальные консультации); взаимодействие в электронной информационно-образовательной среде вуза	130
Промежуточная аттестация обучающихся – Зачет с оценкой	4

**5 Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебной работы**

Таблица 3 – Структура и содержание дисциплины (модуля)

Наименование разделов, тем и содержание материала	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах)			СРС
	Контактная работа преподавателя с обучающимися			
	Лекции	Семинарские (практические занятия)	Лабораторные занятия	
<b>Многопроцессорные вычислительные системы</b>				
<b>Системы параллельной обработки данных и архитектура процессоров</b> <i>Введите содержание материала</i> Виды вычислительных систем. Системы параллельной обработки данных. Конвейер и матрица. Классификация Флинна. Пути повышения производительности: законы Мура, Гроша, Амдала и гипотеза Минского. Архитектуры с сокращенным набором команд, со сверхдлинным командным словом. Векторные архитектуры. Многопроцессорные синхронные вычислительные системы. Конвейерные системы. Матричные системы. Систолические системы.	0.5			

<b>Параллельное выполнение процессов</b> <i>Введите содержанПонятие процесса. Создание параллельных процессов и работы с ними в операционной системе Windows средствами языка программирования C++. Использование объектов синхронизации для обмена данными между процессами.иe материала</i>	0.5			
<b>Программирование параллельных вычислительных процессов</b> <i>Программная реализация создания и работы двух параллельных процессов. Синхронизация работы параллельных процессов с помощью средств ; операционной системы Windows.</i>			1	
-				25
<b>Разработка многопоточных приложений</b>				
<b>Загрузка и выполнение параллельных потоков</b> <i>Понятие потока. Средства языка программирования C++ для создания потоков и работы с ними. ; Создание многопоточных программ и</i>	0.5			
<i>применение семафоров в операционной системе Windows.</i>				
<b>Разработка многопоточных программ</b> <i>Распараллеливание рекурсивных подпрограмм. Преобразование рекурсивных подпрограмм в многопоточные приложения. Многопоточный метод сдваивания.</i>	0.5			
<b>Методы распараллеливания рекурсивных подпрограмм</b>			1	
-				25
<b>Классические задачи синхронизации</b>				

<b>Семафоры и события</b> <i>Проблема взаимного исключения. Постановки классических задач синхронизации и методы их решения. Понятие семафора. Понятие события. Алгоритм Деккера для решения задач синхронизации. Алгоритм Петерсона для решения задач синхронизации. Применение семафоров для решения задачи сериализации. Разработка многопоточных приложений с использованием событий.</i>	1			
<b>Решения проблемы сериализации с помощью семафоров</b> <i>Разработка многопоточного приложения, в котором синхронизация работы потоков осуществляется с помощью семафоров.</i>			1	
-				25
<b>Моделирование параллельных вычислений</b>				
<b>Сети Петри и их применение. Волновые системы</b> <i>Определение и примеры сетей Петри. Асинхронные системы переходов. Расслоенные критические секции. Задача о читателях и писателях. Задача о производителе и потребителе. Задача о философях и ситуация отталкивания. Использование сетей Петри и волновых систем для моделирования работы параллельных программ. &amp;nbsp; Построение параллельных программ с помощью сетей Петри. Разработка класса-канала на основе решения задачи о производителе и потребителе.</i>	1			
<b>Построение и программирование волновых систем с помощью сетей Петри</b>			3	
-				17
<b>Расчетно-графическая работа</b> <i>Выполнение и оформление отчета к расчетно-графической работе</i>				38
<b>Промежуточная аттестация 4 часа</b>				
<b>ИТОГО по дисциплине</b>	4		6	130

**6 Внеаудиторная самостоятельная работа обучающихся по дисциплине (модулю)**



При планировании самостоятельной работы студенту рекомендуется руководствоваться следующим распределением часов на самостоятельную работу (таблица 4):

Таблица 4 – Рекомендуемое распределение часов на самостоятельную работу

<b>Компоненты самостоятельной работы</b>	<b>Количество часов</b>
Изучение теоретических разделов дисциплины	32
Выполнение отчета и подготовка к защите лаб. раб.	60
Выполнение отчета и подготовка к защите РГР	38

## **7 Оценочные средства для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине (модулю)**

Фонд оценочных средств для проведения текущего контроля успеваемости и промежуточной аттестации представлен в Приложении 1.

Полный комплект контрольных заданий или иных материалов, необходимых для оценивания результатов обучения по дисциплине (модулю), практике хранится на кафедреразработчике в бумажном и электронном виде.

## **8 Учебно-методическое и информационное обеспечение дисциплины (модуля)**

### **8.1 Основная литература**

1 Богачев, К. Ю. Основы параллельного программирования : учебное пособие / К.

Ю. Богачёв . – 4-е изд., электрон. – М. : Лаборатория знаний, 2020. – 345 с. // ZNANIUM.COM : электронно–библиотечная система. – Режим доступа: <https://znanium.com/catalog/php?>

2 Борзунов, С. В. Практикум по параллельному программированию: Учебное пособие / Борзунов С.В. – СПб:БХВ-Петербург, 2017. – 236 с. // ZNANIUM.COM : электронно– библиотечная система. – Режим доступа: <https://znanium.com/catalog/php?>

3 Воеводин, В. В. Параллельные вычисления: Учебное пособие / Воеводин В.В., Воеводин В.В. – СПб:БХВ-Петербург, 2015. – 603 с. // ZNANIUM.COM : электронно–библиотечная система. – Режим доступа: <https://znanium.com/catalog/php?>

4 Колдаев, В. Д. Основы алгоритмизации и программирования [Электронный ресурс]: учебное пособие / В.Д. Колдаев; под ред. Л.Г. Гагариной. – М.: ИД ФОРУМ: ИНФРА–М, 2015. – 416 с. // ZNANIUM.COM : электронно– библиотечная система. – Режим доступа:

<http://www.znanium.com/catalog.php?>

### **8.2 Дополнительная литература**

1 Батчер, П. Семь моделей конкуренции и параллелизма за семь недель. Раскрываем тайны потоков / П. Батчер; пер. с англ. А. Н. Киселев. – М.: ДМК Пресс, 2015. – 360с.

2 Гома, Х. UML. Проектирование систем реального времени, параллельных и распределенных приложений / Х. Гома; Пер. с англ. А.А.Слинкина. – М.: ДМК Пресс, 2016. – 698с. – (Объектно-ориентированные технологии в программировании).

3 Грегори, К. С++ AMP: построение массивно параллельных программ с помощью Microsoft Visual C++ / К. Грегори, Э. Миллер. – М.: ДМК Пресс, 2013.

4 Максимов, Н. В. Архитектура ЭВМ и вычислительных систем [Электронный ресурс]: учебник / Н.В. Максимов, Т.Л. Партыка, И.И. Попов. – 5–е изд., перераб. и доп. – М. : Форум: НИЦ ИНФРА–М, 2015. – 512с. // ZNANIUM.COM : электронно–библиотечная система. – Режим доступа: <http://www.znanium.com/catalog.php?>.

8.3 Современные профессиональные базы данных и информационные справочные системы, используемые при осуществлении образовательного процесса по дисциплине

1. Электронно-библиотечная система ZNANIUM.COM. Договор ЕП 44 № 003/10 эбс ИКЗ 191272700076927030100100120016311000 от 17 апреля 2019 г.
2. Электронно-библиотечная система IPRbooks. Лицензионный договор № ЕП44 № 001/9 на предоставление доступа к электронно-библиотечной системе IPRbooks ИКЗ 191272700076927030100100090016311000 от 27 марта 2019 г.
3. Электронно-библиотечная система eLIBRARY.RU. Договор № ЕП 44 № 004/13 на оказание услуг доступа к электронным изданиям ИКЗ 91272700076927030100100150016311000 от 15 апреля 2019 г.
4. Информационно-справочные системы «Кодекс»/ «Техэксперт». Соглашение о сотрудничестве № 25/19 от 31 мая 2019 г.

8.4 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

1 Информационно-аналитический центр по параллельному программированию <http://www.parallel.ru> . 2 Национальный открытый университет ИНТУИТ <http://www.intuit.ru>.

8.5 Лицензионное и свободно распространяемое программное обеспечение, используемое при осуществлении образовательного процесса по дисциплине

Таблица 5 – Перечень используемого программного обеспечения

Наименование ПО	Реквизиты / условия использования
Microsoft Imagine Premium	Лицензионный договор АЭ223 №008/65 от 11.01.2019
OpenOffice	Свободная лицензия, условия использования по ссылке: <a href="https://www.openoffice.org/license.html">https://www.openoffice.org/license.html</a>

Visual Studio Community 2017	Распространяется свободно, может использовать неограниченное число пользователей в организации в учебных аудиториях / <a href="https://visualstudio.microsoft.com/ru/vs/community">https://visualstudio.microsoft.com/ru/vs/community</a>
------------------------------	---

## **9 Организационно-педагогические условия**

Организация образовательного процесса регламентируется учебным планом и расписанием учебных занятий. Язык обучения (преподавания) - русский. Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут.

При формировании своей индивидуальной образовательной траектории обучающийся имеет право на перезачет соответствующих дисциплин и профессиональных модулей, освоенных в процессе предшествующего обучения, который освобождает обучающегося от необходимости их повторного освоения.

### **9.1 Образовательные технологии**

Учебный процесс при преподавании курса основывается на использовании традиционных, инновационных и информационных образовательных технологий. Традиционные образовательные технологии представлены лекциями и семинарскими (практическими) занятиями. Инновационные образовательные технологии используются в виде широкого применения активных и интерактивных форм проведения занятий. Информационные образовательные технологии реализуются путем активизации самостоятельной работы студентов в информационной образовательной среде.

### **9.2 Занятия лекционного типа**

Лекционный курс предполагает систематизированное изложение основных вопросов учебного плана.

На первой лекции лектор обязан предупредить студентов, применительно к какому базовому учебнику (учебникам, учебным пособиям) будет прочитан курс.

Лекционный курс должен давать наибольший объем информации и обеспечивать более глубокое понимание учебных вопросов при значительно меньшей затрате времени, чем это требуется большинству студентов на самостоятельное изучение материала.

### **9.3 Занятия семинарского типа**

Семинарские занятия представляют собой детализацию лекционного теоретического материала, проводятся в целях закрепления курса и охватывают все основные разделы.

Основной формой проведения семинаров является обсуждение наиболее проблемных и сложных вопросов по отдельным темам, а также разбор примеров и ситуаций в аудиторных условиях. В обязанности преподавателя входят: оказание методической помощи и консультирование студентов по соответствующим темам курса.

Активность на семинарских занятиях оценивается по следующим критериям:

- ответы на вопросы, предлагаемые преподавателем;
- участие в дискуссиях;
- выполнение проектных и иных заданий;

- ассистирование преподавателю в проведении занятий.

Ответ должен быть аргументированным, развернутым, не односложным, содержать ссылки на источники.

Доклады и оппонирование докладов проверяют степень владения теоретическим материалом, а также корректность и строгость рассуждений.

Оценивание заданий, выполненных на семинарском занятии, входит в накопленную оценку.

#### **9.4 Самостоятельная работа обучающихся по дисциплине (модулю)**

Самостоятельная работа студентов – это процесс активного, целенаправленного приобретения студентом новых знаний, умений без непосредственного участия преподавателя, характеризующийся предметной направленностью, эффективным контролем и оценкой результатов деятельности обучающегося.

Цели самостоятельной работы:

- систематизация и закрепление полученных теоретических знаний и практических умений студентов;
- углубление и расширение теоретических знаний;
- формирование умений использовать нормативную и справочную документацию, специальную литературу;
- развитие познавательных способностей, активности студентов, ответственности и организованности;
- формирование самостоятельности мышления, творческой инициативы, способностей к саморазвитию, самосовершенствованию и самореализации;
- развитие исследовательских умений и академических навыков.

Самостоятельная работа может осуществляться индивидуально или группами студентов в зависимости от цели, объема, уровня сложности, конкретной тематики.

Технология организации самостоятельной работы студентов включает использование информационных и материально-технических ресурсов университета.

Контроль результатов внеаудиторной самостоятельной работы студентов может проходить в письменной, устной или смешанной форме.

Студенты должны подходить к самостоятельной работе как к наиважнейшему средству закрепления и развития теоретических знаний, выработке единства взглядов на отдельные вопросы курса, приобретения определенных навыков и использования профессиональной литературы.

#### **9.5 Методические указания для обучающихся по освоению дисциплины**

При изучении дисциплины обучающимся целесообразно выполнять следующие рекомендации:

1. Изучение учебной дисциплины должно вестись систематически.
2. После изучения какого-либо раздела по учебнику или конспектным материалам рекомендуется по памяти воспроизвести основные термины, определения, понятия раздела.

3. Особое внимание следует уделить выполнению отчетов по практическим занятиям и индивидуальным комплексным заданиям на самостоятельную работу.

4. Вся тематика вопросов, изучаемых самостоятельно, задается на лекциях преподавателем. Им же даются источники (в первую очередь вновь изданные в периодической научной литературе) для более детального понимания вопросов, озвученных на лекции.

При самостоятельной проработке курса обучающиеся должны:

- просматривать основные определения и факты;
- повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;
- изучить рекомендованную литературу, составлять тезисы, аннотации и конспекты наиболее важных моментов;
- самостоятельно выполнять задания, аналогичные предлагаемым на занятиях; - использовать для самопроверки материалы фонда оценочных средств.

## **10 Описание материально-технического обеспечения, необходимого для осуществления образовательного процесса по дисциплине (модулю)**

### **10.1 Учебно-лабораторное оборудование**

Таблица 6 – Перечень оборудования лаборатории

Аудитория	Наименование аудитории (лаборатории)	Используемое оборудование
305 5 корпус	Лаборатория научно-исследовательской деятельности, курсового и дипломного проектирования.	8 ноутбуков Lenovo core i3

При реализации дисциплины «Параллельное программирование» на базе профильной организации используется материально-техническое обеспечение, перечисленное в таблице 6.1.

Таблица 6.1 – Материально-техническое обеспечение дисциплины

Стандартное или специализированное оборудование, обеспечивающее выполнение заданий	Назначение оборудования
10 персональных ЭВМ, каждая из которых оснащена процессором Intel(R) Core (TM) i3-2100 CPU @3.10 GHz и оперативной памятью 2ГБ. Операционная система - Windows 7. В классе имеется сетевой коммутатор Cisco catalyst 2960 с ПО IOS ver 12.2(55)SE5.	Работа на лабораторных и практических занятиях

### **10.2 Технические и электронные средства обучения**

Отсутствуют

## **11 Иные сведения**

## **Методические рекомендации по обучению лиц с ограниченными возможностями здоровья и инвалидов**

Освоение дисциплины обучающимися с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах. Предполагаются специальные условия для получения образования обучающимися с ограниченными возможностями здоровья.

Профессорско-педагогический состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии). При необходимости осуществляется дополнительная поддержка преподавания тьюторами, психологами, социальными работниками, прошедшими подготовку ассистентами.

В соответствии с методическими рекомендациями Минобрнауки РФ (утв. 8 апреля 2014 г. N АК-44/05вн) в курсе предполагается использовать социально-активные и рефлексивные методы обучения, технологии социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе. Подбор и разработка учебных материалов производятся с учетом предоставления материала в различных формах: аудиальной, визуальной, с использованием специальных технических средств и информационных систем.

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения (персонального и коллективного использования). Материально-техническое обеспечение предусматривает приспособление аудиторий к нуждам лиц с ОВЗ.

Форма проведения аттестации для студентов-инвалидов устанавливается с учетом индивидуальных психофизических особенностей. Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной или электронной форме (для лиц с нарушениями опорно-двигательного аппарата);
- в печатной форме или электронной форме с увеличенным шрифтом и контрастностью (для лиц с нарушениями слуха, речи, зрения);
- методом чтения ассистентом задания вслух (для лиц с нарушениями зрения).

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге или набором ответов на компьютере (для лиц с нарушениями слуха, речи);
- выбором ответа из возможных вариантов с использованием услуг ассистента (для лиц с нарушениями опорно-двигательного аппарата);
- устно (для лиц с нарушениями зрения, опорно-двигательного аппарата).

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ по дисциплине****«Параллельное программирование»**

Направление подготовки	09.03.01 Информатика и вычислительная техника
Направленность (профиль) образовательной программы	Программное обеспечение средств вычислительной техники и автоматизированных систем
Квалификация выпускника	Бакалавр
Год начала подготовки (по учебному плану)	2020
Форма обучения	Заочная форма
Технология обучения	Традиционная

Курс	Семестр	Трудоемкость, з.е.
4	7	4

Вид промежуточной аттестации	Обеспечивающее подразделение
Зачет с оценкой	Кафедра «Математическое обеспечение и применение ЭВМ»_(Архив)

**1 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций**

Таблица 1 – Компетенции и индикаторы их достижения

Код и наименование компетенции	Индикаторы достижения	Планируемые результаты обучения по дисциплине
Общепрофессиональные		

<p>ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.</p>	<p>ОПК-9.1 Знает методики использования программных средств для решения практических задач  ОПК-9.2 Умеет использовать программные средства для решения практических задач  ОПК-9.3 Владеет навыками использования программных средств для решения практических задач</p>	<p>Знает основные архитектуры параллельных систем, методики использования программных средств для решения практических задач с использованием параллельных алгоритмов  Умеет использовать программные средства для создания программ использующих параллелизм  Владеет навыками программирования для многопроцессорных систем</p>
<p>Профессиональные</p>		
<p>ПК-1 Способен разрабатывать компоненты аппаратно-программных комплексов, сетевых приложений и баз данных, используя современные инструментальные средства и технологии программирования</p>	<p>ПК-1.1 Знает методики и технологии разработки компонентов аппаратно-программных комплексов, сетевых приложений и баз данных на основе использования современных инструментальных средств и технологий программирования  ПК-1.2 Умеет вести разработку компонентов аппаратно-сетевых комплексов, сетевых приложений и баз данных  ПК-1.3 Владеет навыками разработки компонентов аппаратно-программных комплексов, сетевых приложений и баз данных с использованием современных инструментальных средств и технологий программирования</p>	<p>Знает методики и технологии разработки компонентов аппаратно-программных комплексов, сетевых приложений на основе использования современных инструментальных средств и технологий программирования  Умеет вести разработку компонентов аппаратно-сетевых комплексов, сетевых приложений.  Владеет навыками разработки компонентов аппаратно-программных комплексов, сетевых приложений с использованием современных инструментальных средств и технологий программирования</p>

Таблица 2 – Паспорт фонда оценочных средств



Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или ее части)	Наименование оценочного средства	Показатели оценки
Тема1- Системы параллельной обработки данных и архитектура процессоров	ОПК-9, ПК-1	Лабораторная работа 1	Использует знание архитектуры многопроцессорных вычислительных систем при разработке ПО. Умеет оценивать производительность, используя закон Амдала.
Тема2- Параллельное выполнение процессов	ОПК-9, ПК-1	Лабораторная работа 1	Применяет функции WinAPI для создания параллельных процессов и работы с ними. Умеет использовать объекты синхронизации для обмена данными между процессами.
Тема3-Загрузка и выполнение параллельных потоков	ОПК-9, ПК-1	Лабораторная работа 2	Применяет функции WinAPI для создания параллельных потоков и работы с ними. Умеет разрабатывать многопоточные программы на языке C++.
Тема4-Разработка многопоточных программ	ОПК-9, ПК-1	Лабораторная работа 2	Умеет преобразовывать рекурсивные подпрограммы в многопоточные приложения. Умеет разрабатывать программы на C++ с использованием многопоточного метода сдвигания и многопоточного метода ветвей и границ.

Тема5-Семафоры и события	ОПК-9, ПК-1	РГР	Умеет применять алгоритмы Деккера и Петерсона для решения задач синхронизации. Умеет применять семафоры для решения задачи сериализации. Умеет разрабатывать многопоточные приложения с использованием событий.
Тема6-Сети Петри и их применение. Волновые системы	ОПК-9, ПК-1	Лабораторная работа 3	Умеет использовать сети Петри и волновые системы для моделирования работы параллельных программ. Владеет навыками разработки многопоточных программ, применяя семафоры и события для синхронизации работы потоков.

## 2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, представлены в виде технологической карты дисциплины (таблица 3).

Таблица 3 – Технологическая карта

	Наименование оценочного средства	Сроки выполнения	Шкала оценивания	Критерии оценивания
Седьмой семестр				
<i>Промежуточная аттестация в форме зачета с оценкой</i>				
1.	Тест	3-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные

2.	Лабораторная работа 1	3-ая неделя	16 баллов	16–лабораторная работа выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент <i>показал отличные владения навыками параллельного программирования, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями,</i> 10–лабораторная работа выполнена с замечаниями, студент <i>показал хорошее владение навыками параллельного программирования,</i> не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 5- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент <i>показал удовлетворительное владение навыками параллельного программирования,</i> 0- задание не выполнено.
3.	Тест	5-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные
4.	Лабораторная работа 2	5-ая неделя	14 баллов	14–лабораторная работа выполнена полностью, правильно, своевременно, даны полные ответы на дополни-

	<b>Наименование оценочного средства</b>	<b>Сроки выполнения</b>	<b>Шкала оценивания</b>	<b>Критерии оценивания</b>
				тельные вопросы во время защиты работы, студент <i>показал отличные владения навыками параллельного программирования, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями,</i> 8–лабораторная работа выполнена с замечаниями, студент <i>показал хорошее владение навыками параллельного программирования,</i> не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 4- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы,

				студент показал удовлетворительное владение навыками <i>параллельного программирования</i> , 0- задание не выполнено.
5.	Тест	9-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные
6.	РГР	9-ая неделя	28 баллов	28– РГР выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент <i>показал отличные умения разработки многопоточных приложений и владения навыками параллельного программирования, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями</i> , 16 – РГР выполнена с замечаниями, студент <i>показал хорошее умения разработки многопоточных приложений и владения навыками параллельного программирования</i> , но не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 7- студент выполнил работу с существенными неточностями, не соблюдены сроки выполнения работы, студент <i>показал удовлетворительное умения разработки многопоточных приложений и владения навыками параллельного программирования</i> , 0- задание не выполнено.
7.	Тест	12-ая неделя	4 балла	4 – правильные ответы на все вопроса, 3- правильные ответы на три вопроса, 2- правильные ответы на два вопроса, 1 – правильный ответ на один вопрос, 0 - все ответы неправильные

8.	Лабораторная работа 3	12-ая неделя	26 баллов	26– РГР выполнена полностью, правильно, своевременно, даны полные ответы на дополнительные вопросы во время защиты работы, студент <i>показал отличные умения разработки многопоточных приложений и владения навыками параллельного программирования, отчет выполнен аккуратно и в соответствии с предъявляемыми требованиями</i> , 17 – РГР выполнена с замечаниями, студент <i>показал хорошее умения разработки многопоточных приложений и владения навыками параллельного программирования</i> , но не выдержаны сроки выполнения работы, даны неполные ответы на дополнительные вопросы во время защиты работы, 8- студент выполнил работу с существенными неточностями, не
	<b>Наименование оценочного средства</b>	<b>Сроки выполнения</b>	<b>Шкала оценивания</b>	<b>Критерии оценивания</b>
				соблюдены сроки выполнения работы, студент <i>показал удовлетворительное умения разработки многопоточных приложений и владения навыками параллельного программирования</i> , 0- задание не выполнено.
ИТОГО:			100 балла	-
<p><b>Критерии оценки результатов обучения по дисциплине:</b>  0 – 64 % от максимально возможной суммы баллов – «неудовлетворительно» (недостаточный уровень для текущей аттестации по дисциплине);  65 – 74 % от максимально возможной суммы баллов – «удовлетворительно» (пороговый (минимальный) уровень);  75 – 84 % от максимально возможной суммы баллов – «хорошо» (средний уровень); 85 – 100 % от максимально возможной суммы баллов – «отлично» (высокий (максимальный) уровень)</p>				

### Задания для текущего контроля

#### Лабораторная работа 1 «Программирование параллельных вычислительных процессов»

Задание. Разработать программу, запускающую параллельно работающие процессы. Первый процесс выводит графические объекты

согласно варианту (прямоугольники) в левое окно, а второй выводит графические объекты согласно варианту (треугольники) в правое окно.

Варианты заданий приведены в таблице 7.

**Пояснение.** Область вывода состоит из двух окон: верхнего и нижнего в нечетных вариантах или левого и правого в четных вариантах. Каждый процесс выводит в свое окно фигуры заданного вида, у которых случайно изменяются цвет, размеры и положение.

Например, в варианте 15, процессы выводят геометрические фигуры в верхнее и нижнее окна, поскольку 15 – нечетное число. Первый процесс выводит в верхнее окно закрашенные круги. Второй процесс выводит в нижнее окно отрезки, у которых изменяются координаты концов.

Таблица 7 – Варианты заданий

Левое окно \ Правое окно	Прямоуголь- ник	Окружность	Отрезок	Закрашен- ный круг	Закрашенн ый прямо- угольник
Прямоуго- льник	-	1	3	5	7
Окружно- сть	2	-	9	11	13
Отрезок	4	6	-	15	17
Закрашен- ный круг	8	10	12	-	19
Закрашен- ный прямо- угольник	14	16	18	20	-
Эллипс	21	22	23	24	25

## Лабораторная работа 2 «Методы распараллеливания рекурсивных подпрограмм»

Задание . Разработать программу, решающую данную задачу с помощью рекурсивной функции. Преобразовать эту функцию и главную программу в многопоточную программу. Поточковые функции выполняются со случайной задержкой.

### Варианты заданий

$$n \cdot (n-1) \cdot x_{2k-1}$$

1. Вычисление значений многочлена  $\sin(x) = \sum_{k=0}^n \frac{(-1)^k x^{2k+1}}{(2k+1)!}$  по

схеме Горнера методом сдваивания.

2. Перебор последовательностей целых чисел  $0 \leq x_1 \leq x_2 \leq \dots \leq x_m \leq n$  при заданных  $m$  и  $n$ .

3. Вычисление значений многочлена  $\cos(x) = \sum_{k=0}^n \frac{(-1)^k x^{2k}}{(2k)!}$  по схеме

Горнера методом сдваивания.

4. Перебор последовательностей целых чисел  $(x_1, x_2, \dots, x_m)$ , находящихся в диапазоне  $0 \leq x_i \leq n$  и попарно не равных между собой.

5. Вычисление значений многочлена  $\ln(1-x) = -\sum_{k=1}^n (x^k)/k$  по схеме

Горнера методом сдваивания.

6. Перебор неубывающих последовательностей целых чисел  $0 \leq x_1 \leq x_2 \leq \dots \leq x_m \leq n$  при заданных  $m$  и  $n$ .

7. Вычисление значений многочлена  $\cosh(x) = \sum_{k=0}^n \frac{x^{2k}}{(2k)!}$  по схеме Горнера

методом сдваивания.

8. Перебор расстановок восьми ферзей, не угрожающих друг другу.

9. Вычисление значений многочлена  $\sinh(x) = \sum_{k=0}^n \frac{x^{2k+1}}{(2k+1)!}$  по схеме Горнера

методом сдваивания.

10. Перебор последовательностей, состоящих из букв A, B и C, не содержащих смежных идентичных подтекстов и имеющих заданную длину  $n$

11. Вычисление значений многочлена  $\sum_{k=1}^n \binom{n}{k} 2^{k-1} x^k$  по схеме Горнера

методом сдваивания.



12. Перебор разменов монетами заданной суммы денег.
13. Вычисление значений  $x_n$ , заданных с помощью рекуррентных соотношений  $x_0 = a$ ,  $x_n = nx_{n-1} + (n-1)^2$ , методом сдваивания.
14. Обменная поразрядная сортировка.
15. Вычисление значений  $x_n$ , заданных с помощью рекуррентных соотношений  $x_0 = a$ ,  $x_n = n^2 x_{n-1} + n - 1$ , методом сдваивания.
16. Перебор всех последовательностей целых чисел  $(x_0, x_1, \dots, x_{n-1})$ , находящихся в диапазоне  $0 \leq x_i \leq n$ .
17. Вычисление значений  $x_n$ , заданных с помощью рекуррентных соотношений  $x_0 = a$ ,  $x_n = \frac{x_{n-1} + 1}{n-1}$ , методом сдваивания.
18. Закраска вершин графа.
19. Вычисление значений  $x_n$ , заданных с помощью рекуррентных соотношений  $x_0 = a$ ,  $x_n = x_{n-2} + n - 1$ , методом сдваивания.
20. Перебор всех подмножеств конечного множества.
21. Вычисление значений  $x_n$ , заданных с помощью рекуррентных соотношений  $x_0 = a$ ,  $x_n = x_{n-3} + n - 1$ , методом сдваивания.
22. Перебор всех последовательностей целых чисел  $0 \leq x_1 \leq x_2 \leq \dots \leq x_m$ , для которых  $x_1 + x_2 + \dots + x_m = n$ . Числа  $m$  и  $n$  заданы.
23. Вычисление значений  $x_n$ , заданных с помощью рекуррентных соотношений  $x_0 = a$ ,  $x_n = \overline{x_{2^{n-1}} + n}$ , методом сдваивания.
24. Перебор всех последовательностей положительных чисел  $(x_1, x_2, \dots, x_{2n})$ , удовлетворяющих неравенствам  $x_1 \leq x_2 \leq x_3 \leq x_4 \leq \dots \leq x_{2n}$  и равенствам  $x_1 + x_2 + \dots + x_{2n} = m$ . Числа  $n$  и  $m$  заданы.
25. Вычисление значений  $x_n$ , заданных с помощью рекуррентных

соотношений  $x_0 \in [a, x_n \in 2^{x_{n-1}} \in n]$ , методом сдвигания.  $n \in [1, n]$

### Расчетно-графическая работа «Решения проблемы сериализации с помощью семафоров»

Задание. Разработать многопоточную программу для вычисления заданного логического выражения, состоящего из поразрядных операций, арифметического выражения или приближенного значения интеграла. Недостающие данные можно вводить с помощью генератора случайных чисел.

Программа должна запускать  $n$  потоков. Параллельные вычисления в потоках производятся с небольшими случайными задержками.

Проверить результат вычисления с помощью подпрограммы, состоящей из одного цикла.

#### Варианты заданий

1.  $(a_0 \oplus b_0)(a_1 \oplus b_1) \dots (a_{n-1} \oplus b_{n-1})$ .
2.  $a_0 \& b_0 \oplus a_1 \& b_1 \oplus \dots \oplus a_{n-1} \& b_{n-1}$ .
3.  $\sin(a_0) \oplus \sin(a_1) \oplus \dots \oplus \sin(a_{n-1})$ .
4.  $\int_0^n \sin(x) dx \oplus \int_{kn}^{kn+1} \sin(x) dx \oplus \dots \oplus \int_{(n-1)k}^{nk} \sin(x) dx$ .
5.  $\min(a_0 \oplus b_0, a_1 \oplus b_1, \dots, a_{n-1} \oplus b_{n-1})$ .
6.  $a_0 b_0 \oplus a_1 b_1 \oplus \dots \oplus a_{n-1} b_{n-1}$ .
7.  $(a_0 \oplus b_0) \& (a_1 \oplus b_1) \& \dots \& (a_{n-1} \oplus b_{n-1})$ .
8.  $\arctg(a_0) \oplus \arctg(a_1) \oplus \dots \oplus \arctg(a_{n-1})$ .
9.  $\int_a^n \arctg(x) dx \oplus \int_{kn}^{kn+1} \arctg(x) dx \oplus \dots \oplus \int_{(n-1)k}^{nk} \arctg(x) dx$ .
10.  $\max(a_0 \oplus b_0, a_1 \oplus b_1, \dots, a_{n-1} \oplus b_{n-1})$ .
11.  $(a_0 \oplus b_0)(a_1 \oplus b_1) \dots (a_{n-1} \oplus b_{n-1})$ .
12.  $(\oplus a_0 \oplus b_0) \& (\oplus a_1 \oplus b_1) \& \dots \& (\oplus a_{n-1} \oplus b_{n-1})$ .
13.  $\cos(a_0) \oplus \cos(a_1) \oplus \dots \oplus \cos(a_{n-1})$ .
14.  $\int_a^n \exp(x) dx \oplus \int_{kn}^{kn+1} \exp(x) dx \oplus \dots \oplus \int_{(n-1)k}^{nk} \exp(x) dx$ .

$$15. \quad \max(a_0, b_0) \cdot \max(a_1, b_1) \cdot \dots \cdot \max(a_{n-1}, b_{n-1}).$$

 $n-1$ 

$$16. \quad \prod_{k=0}^n (a_k b_k)^k.$$

 $k=0$ 

$$17. \quad \max(\min(a_0, b_0), \min(a_1, b_1), \dots, \min(a_{n-1}, b_{n-1})).$$

$$18. \quad \exp(a_0^2) \cdot \exp(a_1^2) \cdot \dots \cdot \exp(a_{n-1}^2).$$

 $a$ 

$$19. \quad \int_0^x \cos(x) dx = \sin(x) - \sin(0) = \sin(x).$$

$$20. \quad \min(a_0 b_0, a_1 b_1, \dots, a_{n-1} b_{n-1}).$$

 $n-1$ 

$$21. \quad \prod_{k=0}^n (a_k + b_k)^k.$$

 $k=0$  $n-1$        $k$ 

$$22. \quad e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}, \text{ где } x \geq 0.$$

 $k=0$ 

$$23. \quad \max(\sin(a_0), \sin(a_1), \dots, \sin(a_{n-1})).$$

$$24. \quad \frac{dx}{x} = \frac{1}{x} dx = \int \frac{1}{x} dx = \ln|x| + C.$$

 $I = x$  $n$ 

$$25. \quad \sum_{k=1}^n \frac{1}{k^s}, \text{ где } 1 < s < 2.$$

 $k=1$        $k$ 

### Лабораторная работа 3 «Построение и программирование волновых систем с помощью сетей Петри»

Задание. Разработать программу, которая, согласно варианту, загружает потоки, реализующие волновую систему. Привести сеть Петри этой волновой системы.

**Варианты заданий**

1.  $x_n \square \sin(u_n \square v_n)$ ,  $y_n \square \exp(u_n) \square \cos(v_n)$ . 2.  $x_n \square \cos(u_n \square v_n)$ ,  $y_n \square \sin(u_n) \square \exp(v_n)$ .
3.  $x_n \square u_n \square (\sin(u_n) \square \cos(v_n))$ ,  $y_n \square \sin(\exp(v_n))$ .
4.  $x_n \square \exp(\sin(u_n))$ ,  $y_n \square \sin(u_n \square v_n)$ .
5.  $x_n \square \cos(u_n \square v_n)$ ,  $y_n \square \cos(\exp(v_n) \square u_n)$ .
6.  $x_n \square \sin(u_n \square v_n)$ ,  $y_n \square \exp(\sin(u_n \square v_n))$ .
7.  $x_n \square \exp(u_n) \square \cos(v_n)$ ,  $y_n \square \cos(u_n \square v_n)$ .
8.  $x_n \square \sin(u_n) \square \exp(v_n)$ ,  $y_n \square \sin(\cos(v_n))$ .
9.  $x_n \square \cos(\sin(u_n \square v_n))$ ,  $y_n \square \sin(u_n \square v_n)$ .
10.  $x_n \square u_n \square v_n \square \sin(v_n)$ ,  $y_n \square \cos(\sin(\cos(v_n)))$ .
11.  $x_n \square \exp(\cos(u_n))$ ,  $y_n \square (\sin(u_n) \square \cos(v_n)) \square \exp(v_n)$ .
12.  $x_n \square \sin(u_n \square v_n)$ ,  $y_n \square \exp(\sin(u_n \square v_n))$ .
13.  $x_n \square \cos(u_n \square v_n)$ ,  $y_n \square \exp(\cos(u_n \square v_n))$ .
14.  $x_n \square \sin(\cos(u_n))$ ,  $y_n \square \exp(\sin(u_n) \square \sin(v_n))$ .
15.  $x_n \square \sin(u_n) \square \exp(v_n)$ ,  $y_n \square \cos(u_n \square v_n)$ .
16.  $x_n \square \sin(\exp(u_n))$ ,  $y_n \square u_n \square (\sin(u_n) \square \cos(v_n))$ .
17.  $x_n \square \cos(u_n \square v_n)$ ,  $y_n \square \exp(\cos(v_n))$ .
18.  $x_n \square \cos(\exp(u_n) \square v_n)$ ,  $y_n \square \sin(u_n \square v_n)$ .
19.  $x_n \square \exp(\cos(u_n \square v_n))$ ,  $y_n \square \sin(u_n \square v_n)$ .
20.  $x_n \square \cos(u_n \square v_n)$ ,  $y_n \square \exp(u_n) \square \cos(v_n)$ .
21.  $x_n \square \sin(\sin(u_n))$ ,  $y_n \square \sin(u_n) \square \exp(v_n)$ .
22.  $x_n \square \sin(u_n \square v_n)$ ,  $y_n \square \cos(\sin(u_n \square v_n))$ .
23.  $x_n \square \sin(\cos(\sin(u_n)))$ ,  $y_n \square u_n \square v_n \square \cos(v_n)$ .
24.  $x_n \square (\sin(u_n) \square \cos(v_n)) \square \exp(u_n)$ ,  $y_n \square \exp(\cos(v_n))$ .
25.  $x_n \square \exp(\cos(u_n \square v_n))$ ,  $y_n \square \cos(u_n \square v_n)$ .

**Примерная структура билетов тестирования****Тест «Программирование параллельных вычислительных процессов»**

**Вопрос 1.** Установить соответствие между законами повышения производительности вычислительной системы и их описаниями.

*Законы:* 1) закон Мура; 2) закон Гроша; 3) гипотеза Минского; 4) закон Амдала.

*Описания:* 1) если доля последовательно выполняющихся в программе равна  $f$ , то ускорение параллельного вычисления этой программы не превышает  $1/f$ ;

2) в параллельной системе с  $n$  процессорами, производительность каждого из которых равна единице, общая производительность растет как  $\log n$ ;

2

3) производительность одного процессора увеличивается пропорционально квадрату его стоимости;

4) количество транзисторов на кристалле процессора и производительность процессора удваиваются каждые полтора года.

**Вопрос 2.** Установить соответствие между функциями классами классификации Флинна и примерами вычислительных систем, принадлежащим к этим классам.

*Функции:* 1) SISD; 2) SIMD); 3) MISD; 4) MIMD с локальной памятью; MIMD с общей памятью.

*Назначение:* 1) симметричная мультипроцессорная система SMP;

2) система с массовым параллелизмом MPP;

3) конвейерная система;

4) матричный процессор; 5) обычный компьютер.

**Вопрос 3.** Установить соответствие между функциями Windows API и их назначением.

*Функции:* 1) CreateProcess();

2) CreateThread(); 3)

TerminateProcess();

4) TerminateThread().

*Назначение:* 1) принудительное завершение потока;

2) принудительное завершение процесса;

3) загрузка потока;

4) загрузка процесса.

**Вопрос 4.** Какое расширение имеет загружаемый процесс:

**Варианты ответа:**

1. exe;                      2. dll;

2. cpp;                      4. com.

**Тест «Методы распараллеливания рекурсивных подпрограмм»**

**Вопрос 1.** Какой из приведенных ниже прототипов функции загружаемого потока описан верно?

**Варианты ответа:**

- 1) DWORD WINAPI thr(LPVOID p);
- 2) int thr(void \*p);
- 3) int thr(int a, int b);
- 4) DWORD WINAPI thr(int a).

**Вопрос 2.** Каким образом передаются аргументы подпрограммы загружаемого потока?

**Варианты ответа:**

- 1) определяют структуру, состоящую из аргументов подпрограммы и передают эти аргументы с помощью указателя на структуру;
- 2) как аргументы обычной подпрограммы;
- 3) определяют структуру, состоящую из аргументов подпрограммы и передают эту структуру как аргумент подпрограммы потока; 4) используют дополнительные средства операционной системы.

**Вопрос 3.** Каким образом распараллеливается вычисление ассоциативной операции над элементами массива.

**Варианты ответа:**

- 1) определяется рекурсивная подпрограмма, вычисляющая результат операции над элементами массива, затем рекурсивные вызовы в подпрограмме заменяются на загрузку потока подпрограммы, в главной программе подпрограмма вызывается без загрузки потока;
- 2) определяется рекурсивная подпрограмма, вычисляющая результат операции над элементами массива, затем рекурсивные вызовы в подпрограмме и в главной программе заменяются на загрузку потока подпрограммы;
- 3) определяется рекурсивная подпрограмма, вычисляющая результат операции над элементами массива, затем в подпрограмме потока оставляется рекурсивный вызов а в главной программе осуществляется загрузка потока подпрограммы.

**Тест «Решения проблемы сериализации с помощью семафоров»**

**Вопрос 1.** Установить соответствие между функциями Windows API и их назначением.

*Функции:* 1) CreateSemaphore(); 2) CreateMutex(); 3) CreateEvent().

*Назначение:* 1) создание объекта события; 2) создание бинарного семафора; 3) создание семафора.

**Вопрос 2.** Установить соответствие между функциями Windows API и их назначением при работе с семафорами.

*Функции:* 1) WaitForSingleObject (); 2) ReleaseSemaphore (). *Назначение:*

1) освобождение семафора; 2) захват семафора.

**Вопрос 3.** Решение какой задачи должно быть использовано для построения многопоточного приложения вычисляющего сумму элементов массива.

**Варианты ответа:**

- 1) проблема взаимного исключения; 2) задача о читателях и писателях;
- 3) задача о философях.

### **Тест «Построение и программирование волновых систем с помощью сетей Петри»**

**Вопрос 1.** В каком из перечисленных ниже случаев срабатывание перехода сети Петри возможно и приводит к маркировке  $M'$ ? **Варианты ответа:**

- 1)  $M \geq \text{pre}(t)$ ,  $M' = M - \text{pre}(t) + \text{post}(t)$ ;
- 2)  $M \leq \text{pre}(t)$ ,  $M' = M - \text{pre}(t) + \text{post}(t)$ ;
- 3)  $M \geq \text{post}(t)$ ,  $M' = M - \text{post}(t) + \text{pre}(t)$ ;
- 4)  $M > \text{pre}(t)$ ,  $M' = M - \text{pre}(t) + \text{post}(t)$ .

**Вопрос 2.** Решение какой задачи используется для построения класса канала?

**Варианты ответа:**

- 1) задача о производителе и потребителе;
- 2) задача о читателях и писателях;
- 3) задача о философях.

**Вопрос 3.** Конвейерная система состоит из  $n$  потоков, соответствующих ее операциям. Синхронизация работы потоков осуществляется с помощью событий. Пусть  $n$  больше 10. Какое число  $k$  наиболее точно приближает количество событий, необходимых для синхронизации? **Варианты ответа:**

1)  $2n$ ; 2)  $n$ ; 3)  $3n$ ; 4)  $4n$ .

**Вопрос 4.** Какое количество каналов необходимо для вычисления значений функции  $z_n = \exp(x_n) + x_n u_n$  с помощью волновой системы?

**Варианты ответа:**

1) 5; 2) 1; 3) 2; 4) 3; 5) 4.



**Лист регистрации изменений к РПД**

№ п/п	Содержание изменения/основание	Кол-во стр. РПД	Подпись автора РПД
1			
2			
3			
4			
5			